UNIVERSITY OF CALIFORNIA SAN DIEGO

Co-Creativity and AI Ethics

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Computer Science

by

Vignesh Gokul

Committee in charge:

Professor Manmohan Chandraker, Co-Chair
Professor Shlomo Dubnov, Co-Chair
Professor Taylor Berg-Kirkpatrick
Professor Sanjoy Dasgupta
Professor Julian McAuley

2024

The Dissertation of Vignesh Gokul is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

To my family and friends.

EPIGRAPH

Creativity is seeing what others see and thinking what no one else ever thought.

Albert Einstein

Artificial intelligence is not a substitute for human intelligence; it is a tool to amplify human creativity and ingenuity.

Fei-Fei Li

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisor Shlomo Dubnov for his continued support throughout my PhD. I would certainly not be who I am now without his guidance. I will always be greatful for all the things I learnt from him over the past 5 years from identifying a problem to writing strong papers. I would also like to thank my committee members Manmohan Chandraker, Julian McAuley, Sanjoy Dasgupta and Taylor Berg-Kirkpatrick for their advising.

Secondly, I would like to thank Christine Alvarado who I am greatly indebted to for making me the person I am today. Thanks to her, I learnt how to manage research groups, educate students and teach classes. I will never forget the three years of ERSP mentoring that I had the good fortune to undertake under her guidance, throughout my life.

My deepest thanks to my wife Parinitha Ganapathisubramaniam Kannan, who stood by my side throughout my PhD. I would not be here today without her total support. Thanks to my family members who motivated me to pursue research.

Special thanks to all my collaborators who made this dissertation possible: Vaikkunth Mugunthan, Gerard Assayag, Ganesh Prassana Balakrishnan and Tammuz Dubnov.

I have also been fortunate enough to receive funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement # 883313, project REACH : Raising Cocreativity in CyberHuman Musicianship)

Chapter 1, contains material in found in Gokul, V., Balakrishnan, G. P., Dubnov, T., & Dubnov, S. (2019, March). Semantic Interaction with Human Motion Using Query-Based Recombinant Video Synthesis. In 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 379-382). IEEE. The dissertation author was the primary investigator and author of the paper.

Chapter 2, is a reprint of material as it appears in Dubnov, S., Assayag, G., & Gokul, V. (2022, August). Creative Improvised Interaction with Generative Musical Systems. In 2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 121-126). IEEE. The dissertation author was one of the primary investigator and author

of the paper. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement #883313, project REACH : Raising Cocreativity in CyberHuman Musicianship)

Chapter 3, is a reprint of material as it appears in Dubnov, S., Gokul, V., & Assayag, G. (2023, February). Switching Machine Improvisation Models by Latent Transfer Entropy Criteria. In Physical Sciences Forum (Vol. 5, No. 1, p. 49). MDPI. The dissertation author was one of the primary investigator and author of the paper. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement #883313, project REACH : Raising Cocreativity in CyberHuman Musicianship)

Chapter 4, is a reprint of material as it appears in Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021, April). Dpd-infogan: Differentially private distributed infogan. In Proceedings of the 1st Workshop on Machine Learning and Systems (pp. 1-6). The dissertation author was one of the primary investigator and author of the paper.

Chapter 5, in part is currently being prepared for submission for publication of the material. Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021). Bias-free fedgan: A federated approach to generate bias-free datasets. arXiv preprint arXiv:2103.09876. The dissertation author was one of the primary investigator and author of the paper.

Chapter 6, in part is currently being prepared for submission for publication of the material. Gokul, V., & Dubnov, S. (2024), PosCUDA: Position based Convolution for Unlearnable Audio Datasets. arXiv preprint arXiv:2401.02135. The dissertation author was the primary investigator and author of the paper.

VITA

2017	Bachelor of Engineering, Anna University, Chennai, India
2019	MS in Computer Science, University of California San Diego
2024	PhD in Computer Science, University of California San Diego

PUBLICATIONS

Gokul, V., Balakrishnan, G. P., Dubnov, T., & Dubnov, S. (2019, March). Semantic Interaction with Human Motion Using Query-Based Recombinant Video Synthesis. In 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 379-382). IEEE.

Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021, April). Dpd-infogan: Differentially private distributed infogan. In Proceedings of the 1st Workshop on Machine Learning and Systems (pp. 1-6).

Dubnov, S., Gokul, V., & Assayag, G. (2023, February). Switching Machine Improvisation Models by Latent Transfer Entropy Criteria. In Physical Sciences Forum (Vol. 5, No. 1, p. 49). MDPI.

Mugunthan, V., Lin, E., Gokul, V., Lau, C., Kagal, L., & Pieper, S. (2022, October). Fedltn: Federated learning for sparse and personalized lottery ticket networks. In European Conference on Computer Vision (pp. 69-85). Cham: Springer Nature Switzerland.

Dubnov, S., Assayag, G., & Gokul, V. (2022, August). Creative Improvised Interaction with Generative Musical Systems. In 2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 121-126). IEEE.

Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021). Bias-free fedgan: A federated approach to generate bias-free datasets. arXiv preprint arXiv:2103.09876.

Gokul, V., & Dubnov, S. (2024), PosCUDA: Position based Convolution for Unlearnable Audio Datasets. arXiv preprint arXiv:2401.02135

ABSTRACT OF THE DISSERTATION

Co-Creativity and AI Ethics

by

Vignesh Gokul

Doctor of Philosophy in Computer Science

University of California San Diego, 2024

Professor Manmohan Chandraker, Co-Chair
Professor Shlomo Dubnov, Co-Chair

With the development of intelligent chatbots, humans have found a method to communicate with artificial digital assistants. However, human beings are able to communicate an enormous amount of information without ever saying a word, eg gestures and music. My research objective is to enable non-verbal communication between humans and artificial agents, a problem I call co-creativity. My work explores controlling generative models that can intelligently interact with data such as gestures (videos) and music (audio).

Another area of my research focuses on ethical issues stemming from using data to train such machine learning models. We investigate ways to protect the privacy of the data owner and prevent unauthorized usage/ leakage of private information during training machine learning models. We then propose a method to create unlearnable audio datasets to prevent unauthorized usage of data for model training.

Introduction

Over the recent years, generative models have shown tremendous capability to generate videos, audios and other forms of multimedia such as deepfakes. This includes solving tasks such as continuing video generation from a single image, video in-painting, audio generation given lyrics. With the development of intelligent chatbots, humans have found a method to communicate with artificial digital assistants. However, human beings are able to communicate an enormous amount of information without ever saying a word. The expressions on our faces, the way we stand, and the gestures we use are all forms of non-verbal communication that can often express information better and faster than anything we say. In fact, non-verbal communication is such an important and integral part of conveying the true meaning of verbal communication that any interaction with a virtual person will not seem natural or human without it. We also communicate and express ourselves with other forms of expressions such as music, art and dance. We introduce a different type of generative problem: modelling interaction strategy between two processes.

My research objective is to enable a non-verbal communication between humans and artificial agents, a problem I call co-creativity. For instance, in the problem of music improvisation, two expert musicians communicate with each other via their music. Similarly, we have other art forms of expression such as dance. Can we simulate such interactive non-verbal communication amongst humans and learning systems? My work also spans to deploying such creative agents for real-time use cases.

While generative models have been deployed for impressive applications, training models such as Large Language Models (LLMs) and other transformer architectures require high compute and large amounts of data. To overcome this, we leverage existing pre-trained models. Pre-trained models are trained on large amounts of data and provide a strong representation that reflects the underlying structure of the data. Now any co-creativity problem such as music/gesture improvisation boils down to: controlling pre-trained models meaningfully based on an input control signal.

Recent deep learning models also depend on large datasets which might be hard to obtain. Usually, model developers scrape the internet for data to be cleaned later. Often this includes blogs, news articles and other content on social media. There are two main ethical issues that stem from such practices: 1) Data scraped from the web such as images/ transactional data might contain sensitive private information. 2) Training models on data sourced from online content might lead to copyright infringement as the model is trained on unauthorized data without the permission of the content owner.

To solve the first problem, I introduce DPD-InfoGAN, a framework to train generative models on private sensitve data in a safe way with privacy guarantees. Using our framework, it is not possible for the trained model to leak private information from the training dataset. To protect unauthorized model training from data scraped online, I introduce PosCUDA, a framework to create unlearnable audio datasets. Such unlearnable datasets are robust to any architecture and feature processing. This means that if an attacker scrapes the protected data online and train a model, the model would not generalize to any data as it would not learn useful features from the protected training dataset.

0.0.1 Dissertation Organization

In this dissertation, I provide an overview of my work on Co-creativity and AI ethics. Chapters 1,2 and 3 discusses my research on co-creativity applications such as gesture bot and musical improvisation systems, while chapters 4,5 and 6 discusses ethical issues from training machine learning systems and solutions to ethically train such models.

In Chapter 1, I discuss learning interaction strategies for co-creativity applications in dance. I elaborate more on the algorithm used (Variable Markov Oracle) and discuss a gesture bot application.

In Chapter 2, I provide a high level overview of musical improvisation systems, the different types of reduced representation to use to represent complex musical signals meaningfully in low-dimensional space. I also discuss a general template architecture for musical improvisation systems.

In Chapter 3, I extend the co-creativity problem to music improvisation. How can we select meaningful musical responses to an input musical prompt? I provide an overview of our measure based on transfer entropy and methods to estimate the transfer entropy between two processes.

In Chapter 4, I discuss methods to train generative models on sensitive/private data. In Chapter 5, I discuss ethical issues that arise from training generative models on distributed setting. I also discuss my work on overcoming such problems and train generative models that are safe for deployment.

Finally in Chapter 6, I discuss PosCUDA, my work on unlearnable audio datasets. Using PosCUDA, I demonstrate how we can achieve unlearnablity for audio classification tasks.

Chapter 1

Semantic Interaction with Human Motion Using Query-Based Recombinant Video Synthesis.

The ability of a machine to understand the motion and behaviour of a particular actor is a very important task in machine vision. This problem has so many possible applications in domains such as motion retargeting, robot navigation, healthcare, psychology, augmented reality applications such as games etc.In this chapter we demonstrate a human-robot interaction system based on a gestural query, where the computer response is a computer generated video of another human movement. This work differs from other recent video retargeting systems since it is not meant to modify the target video as such, but rather query a video database for the most responsive segment through gestural interpretation process. For this purpose we developed a generative video system capable of extracting the latent representation of free movements such as dance and expressive gesture, and querying and re-editing multiple found video segments in response to an input movement query. One of the main challenges in this approach is finding the ”units” of continuous movement input so that both the style of the target video and the relevant aspect of the query video would be related in a meaningful way. In this chapter we describe a gestural motif extraction system that combines deep feature learning with structural similarity analysis to allow such query based human-computer motion interaction. With the surge of deep learning, many models have been proposed to deal with the problem of Learning from Demonstrations or Imitation Learning. The main idea of this work is to have an agent mimic movement of humans in order to create a meaningful response to a user's gesture. This is a difficult problem to solve as there are complex dynamics that need to be captured by the model. Moreover, we are interested in free or abstract movements, such as dance and expressive gestures, in which case the ”meaning” of the gestures can not be predetermined. Moreover, the meaning itself can be self-referential, such as repeated movements in dance or idiomatic gesticulations, or emergent from interaction between two people, in which case each of the participants needs to be able to perceive and interpret the gestures of the other. Several existing works propose features that can be used to model human body, such as PoseNet [124] provides a representation of the skeletal structure. These features are useful in motion re-targeting and this can be done simply by superimposing an image model of the agent body onto the skeleton that is extracted from the input video. However, such techniques are not sufficient for our purpose as our goal is not motion re-targeting, but rather a chat bot like interaction. A problem with using pose net features for our purpose is that it does not take into account the style of the agent or its relation to the input movement. One of the main challenges of our work is finding the ”units” of gesture, so that both the style of the target will be preserved, and that the relevant segments of the input gesture would be used to trigger the agent response in a meaningful way so as to establish a sort of dialog or call-and-response action. In such setting, the freedom in the agent can vary between mirroring and improvisation by controlling the level of imitation versus autonomy.

Deep Neural Networks[96, 52] have showed enormous potential in learning representations of data in a lower dimensional manifold. In this chapter, we use feature learning and sequence modeling to effectively represent human figure and the style of its movement in a video, and efficiently query and retrieve segments (motifs) of video frames from a database to create a new video sequence that mimics the query. Our results show that our system learns deep features that enable the agent to perform almost similar actions to the query. The Autoencoder encodes the images into an embedding space such that it's motion semantics are encaptured. The Variable Markov Oracle (VMO) uses a Viterbi-like dynamic programming algorithm to efficiently choose sub-sequences of action and hence performs guided video synthesis. Modeling complex movement dynamics requires both feature learning and sequential analysis. Consequently, our query-based retrieval system consists of the following modules:

- A feature learning module - to represent the character in each frame in a lower dimensional embedding space such that the motion features are captured.
- A motif extraction module - to represent salient repetitive segments in the target database, based on a distance measure that describes how close two images are in the feature space.
- A querying mechanism - to query and retrieve segments of target movement from the database based on input video.

1.1 Methodology

The feature extractor is the most important module in the system and a failure in extraction of meaningful features will lead to a collapse in detection of motifs and the query mechanism. It is very crucial to train a feature extractor capable of learning robust features that can be used to describe human postures and gestures. The dataset has no supervision and manually labeling frames that are similar to each other in terms of gesture or motion is a cumbersome process. A feasible solution to this problem is to use an unsupervised learning technique to generate meaningful video embeddings.

We trained a Convolutional Autoencoder to extract features by performing pixel-wise reconstruction of the input. The network is shown below (fig. 1.1). The encoder has an architecture similar to that of AlexNet[66] with a 1x1 convolution at the end that outputs 675-dimensional latent space (a $15 \times 15 \times 3$ image) which will serve as a meaningful representation of the data in a lower dimensional manifold. The decoder mirrors the encoder except that it uses fractionally-strided convolutions. To improve the training process, we use an objective that enforces matching between all the encoding and decoding layers.

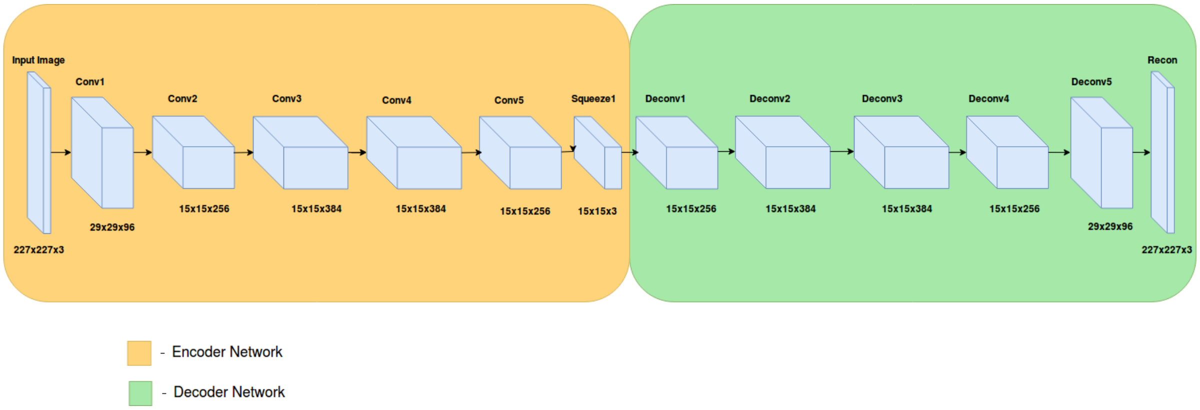


Figure 1.1. The architecture of the network.

$$L = \sum_{k=1}^{5}\sum_{j=1}^{n(k)}\sum_{i=1}^{m(k)} (p_{i,j_{conv_k}} - p_{i,j_{deconv_{6-k}}})^2 + \sum_{j=1}^{227}\sum_{i=1}^{227} (p_{i,j_{input}} - p_{i,j_{recon}})^2 \quad (1.1)$$

In the objective given above, $p_{i,j}$ is the pixel corresponding to the k^{th} convolution and deconvolution or the input and reconstruction(recon) layers shown in the architecture below(fig. 1.1). The values of i and j ranges from 1,1 to m,n depending on the feature map corresponding to the value of k ($conv_k$ and $deconv_k$).

The network was trained on a dataset comprising of images of spiderman dancing, with the background completely green. Since a large portion of pixels in the image are green, the autoencoder learns to reconstruct the background around the person in the frame. Thus the reconstruction is an image with a similar green background but a blurred black outline of the character in the frame. The autoencoder does not focus on the pixel-wise reconstruction of the character, rather learns to capture the semantics of the character's posture. The query before being passed into the encoder needs to be processed in a way such that it is similar to the images in the training data, i.e., needs a green background. For this purpose, a semantic segmentation network[89] trained on the PascalVOC dataset is used to extract the pixels of the foreground character and every other pixel in the image is converted to green.

After the image has been processed, it is fed into the encoder, features extracted and queried using the algorithms described below.

Greedy Method

We initially experimented with a greedy matching technique that chooses the best frame for every frame in the query based on the euclidean distance between the extracted features. To quantitatively see the quality of the video generated we visualize the similarity matrix between the features of our query video and our database video. If the output of the query is continuous, the entry with the lowest values must lie close to a diagonal. Running the greedy algorithm on one of our query videos gave us the similarity matrix shown below (fig. 1.2). The best match for the query frame (rows), is the frame number corresponding to the column associated with the white box in that row. The large similarity matrix shows patches of yellow and white boxes and they are not continuous (thin red lines passing horizontally) and have jumps (vertical lines). The two sections of the similarity matrix that's been zoomed in (10×10 frames) are frames in the query that are very close to each other (distance along y-axis). The circles must be close to each other (distance along x-axis) and the fact that they are not shows that the output is discontinuous. Even within the 10×10 grids, we can see that there are only one or two white boxes, indicating discontinuity. The algorithm we describe in the next section takes care of continuity by creating a data structure based on closeness of frames in the feature space.

Sequence Query using Self-Attention

Our Variable Markov Oracle model can be visualized as a self attention mechanism. Attention[68] has been proposed as a mechanism to facilitate learning of long sequences and is typically used in sequence-to-sequence models. Our query mechanism can be considered as a encoder-decoder architecture, if we have paired corpora. Since our problem is totally unsupervised, we start with the simple ”mirror game” setting. In problems involving complex gestures such as dance or sports, there are long repetitions that need to be modelled and recombined in order to generate plausible movement responses. The pipeline we propose requires two steps: 1) Instantaneous features are extracted using an autoencoder on individual frames. 2) VMO is used to create time embeddings based on clustering similar sub-sequences according

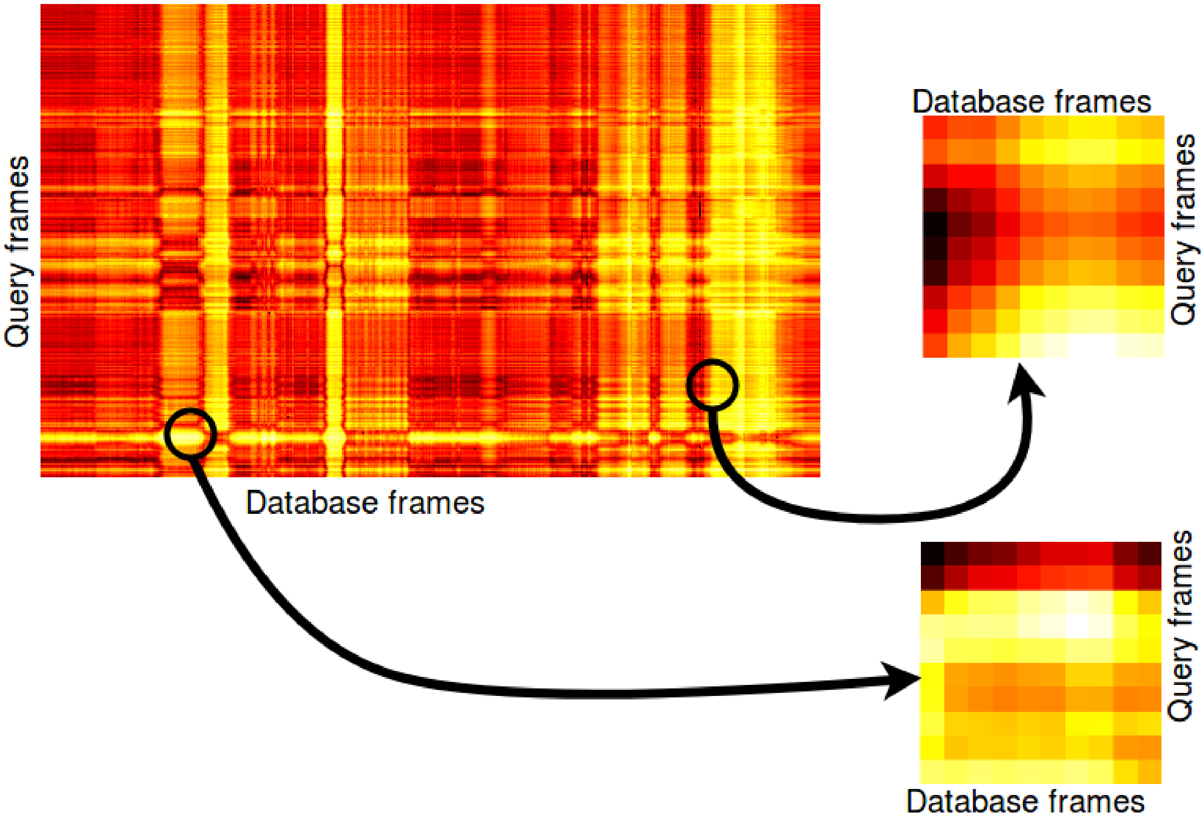


Figure 1.2. Cross-similarity Matrix Visualization between query and database

to common suffix structures. Symbolization of the time-series data is performed by searching over different thresholds of similarity between frames. This is done by capturing self-similarity structures(fig. 1.3) and representing it as a graph of suffix links pointing to locations that are similar upto a certain threshold. This threshold is learned adaptively by maximizing mutual information between each entry on the data sequence and its past [31].

We use the features that are extracted by our feature extraction modules to build an oracle of our dataset that creates suffix links (sfx) for every state possible. The oracle are built based on the euclidean distance between the features. The oracle structure is made up of two kinds of linkages - the forward links (normal arrows) and the suffix links (dashed arrows). The suffix link is a backward pointer that links the state s_t with the state s_k, where $t > k$, and is useful for finding the longest repeated suffix links (lrs). Forward links are of two types - internal, that connects state s_t with the state s_{t+1} and external, that connects state s_t with the state s_{t+k}.

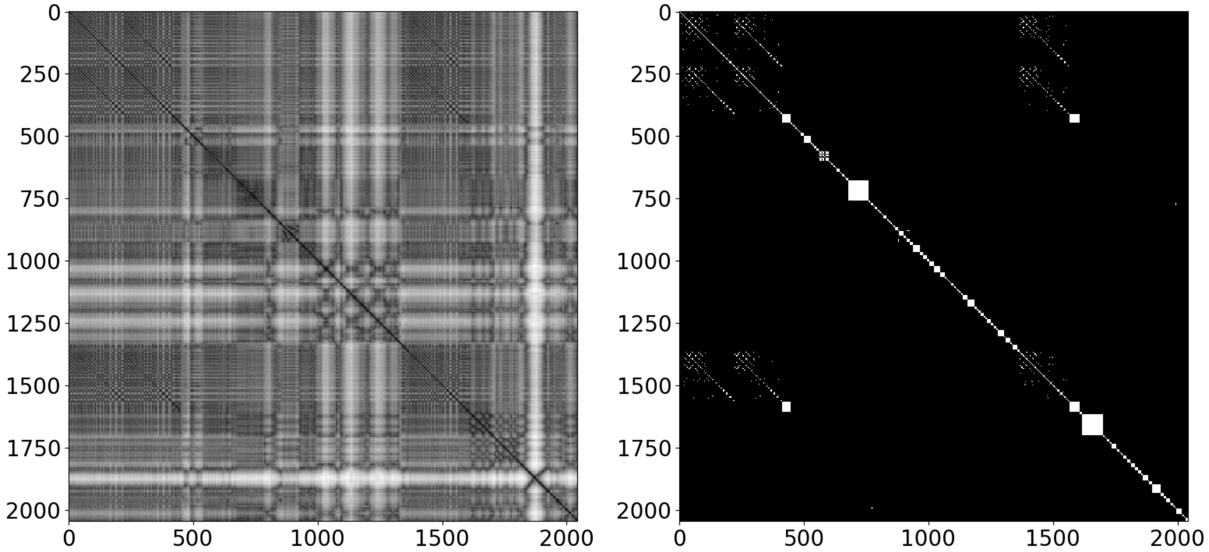


Figure 1.3. The SSM of the original data(left) vs that after the *VMO* has been created(right)

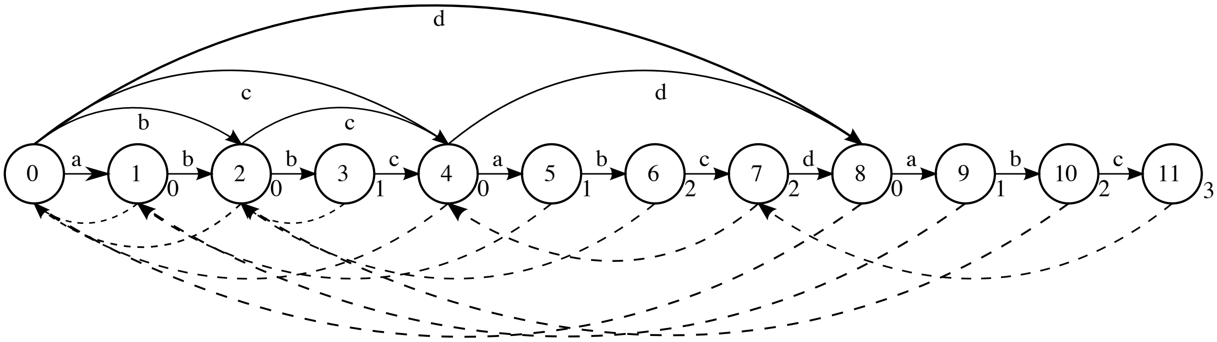


Figure 1.4. The *VMO* data structure

The result for finding repeated patterns in one of the video samples generated is displayed below (fig. 1.5). The y-axis indicates the pattern index of repeated motifs of a signal sampled at discrete times shown along the x-axis. The lines represent repeated motifs, and the long lines show us that the VMO is able to learn motifs accross long intervals of time.

The query matching algorithm (Alg. 2) takes in the query *R* and matches it to the oracle *O* (discussed above), formed by the features extracted from the spiderman video (time series data). The algorithm returns a cost, a corresponding recombination path and the frame index where the query ended. The cost is the reconstruction error between the query and the best match from *O* given a metric on a frame-by-frame basis. The recombination path corresponds

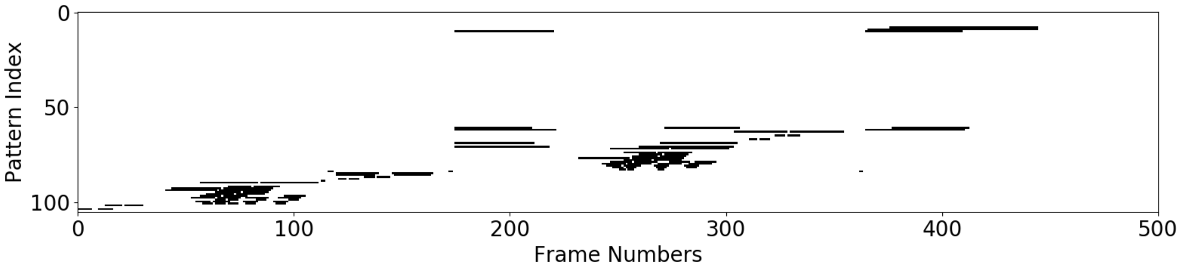


Figure 1.5. Motifs learnt by VMO. The y-axis represents the pattern index. The x-axis represents the frame indices.

Algorithm 1. Distance Greedy (D_{greedy})

```
Require: Target signal in $VMO$, $Oracle(Q = q_1, q_2, \ldots, q_T, O = O[1], O[2], \ldots, O[T])$, query
    frame for time step n $R[N]$
Initialize $d_{min} = \infty$, dist
for $n = 1 : T$ do
    $dist \leftarrow$ Euclidean distance between $R[n]$
        and $O[n]$
    if $dist < d_{min}$ then
        $d_{min} \leftarrow dist$
    end if
end for
return $d_{min}$
```

to the sequence of indices that will reconstruct a new sequence from O that best resembles the query. More details regarding the algorithm can be found in [20, 19]. The problem with this algorithm is that as the length of the query increases there is a higher chance for the response to be repetitive (as it minimizes the recombination cost which corresponds to jump between frames). To overcome this, a break out is initiated from the query if the recombination cost exceeds a certain threshold ($C_{threshold}$). The breaking out is completed if the reconstruction error from VMO (for that particular frame where the breakout was initiated) is more than the best distance for that frame with the entire database, obtained using a greedy approach. Once it breaks out of the VMO a new query is started from the query frame where the previous query broke out. If the break out is not successful (reconstruction error from VMO is equal to best distance from greedy) then we continue with the same query. In other words, we keep appending the path obtained from Alg. 2, check if the index where the VMO broke out is lesser than the length

Algorithm 2. Query matching

```
Require: Target signal in $VMO$, $Oracle(Q = q_1, q_2, \ldots, q_T, O = O[1], O[2], \ldots, O[T])$, query
    time series $R = R[1], R[2], \ldots, R[N]$ and $C_{threshold}$
Get the number of clusters, $M \leftarrow |\Sigma|$
Initialize cost vector $C \in R^M$ and path matrix $P \in R^{M \times N}$.
for $m = 1 : M$ do
    $P_{m,1} \leftarrow$ Find the state, $t$, in the $m$th list from $\Sigma$ with
            the least distance, $d_{m,1}$, to $R[1]$
    $C_m \leftarrow d_{m,1}$
end for
Initialize i = 2
for $n = i : N$ do
    for $m = 1 : M$ do
        $P_{m,n} \leftarrow$ Find the state $t$, in clusters (
            and ) corresponding to forward links
            from state $P_{m,n-1}$ with the least distance,
            $d_{m,n}$ to $R[n]$
        $C_m += d_{m,n}$
        if $C_m > C_{threshold}$ then
            if $D_{greedy}(R[n]) < min(d_{m,n})$ then
                return $P[(C)]$, min($C$), n
            end if
        end if
    end for
end for
return $P[(C)]$, min($C$), n
```

Algorithm 3. Restart mechanism

```
Require: Target signal in $VMO$, $Oracle(Q = q_1, q_2, \ldots, q_T, O = O[1], O[2], \ldots, O[T])$ and query
    time series $R = R[1], R[2], \ldots, R[N]$
Initialize path list $P \in R^N$.
$start = 0$
while $start < N$ do
    $q = R[start], R[start+1], \ldots, R[N]$
    $P_{temp}, c, end \leftarrow$ Query matching (O, q)
    $P[start], P[start+1], \ldots, P[start+end] \leftarrow$
            $P_{temp}[0], P_{temp}[1], \ldots, P_{temp}[end]$
    $start = end$
end while
return P
```

of query, if it is then we restart a new query from that index until we have traversed the entire length of the query. Another problem is that the VMO outputs frames that are not a meaningful response to the query. To fix this, we have to increase the search space and not limit the vmo to just look at the forward links. Choosing the forward links gives a smooth output but not the best match. So we leverage the clustering behaviour of the VMO to choose frames that are similar to the best forward link. This results in a smooth and continuous output. If the $C_{threshold}$ is high it results in the output being smooth and continuous and if $C_{threshold}$ is low then this results in the output trying to imitate the query but with a loss in continuity or smoothness of transition.

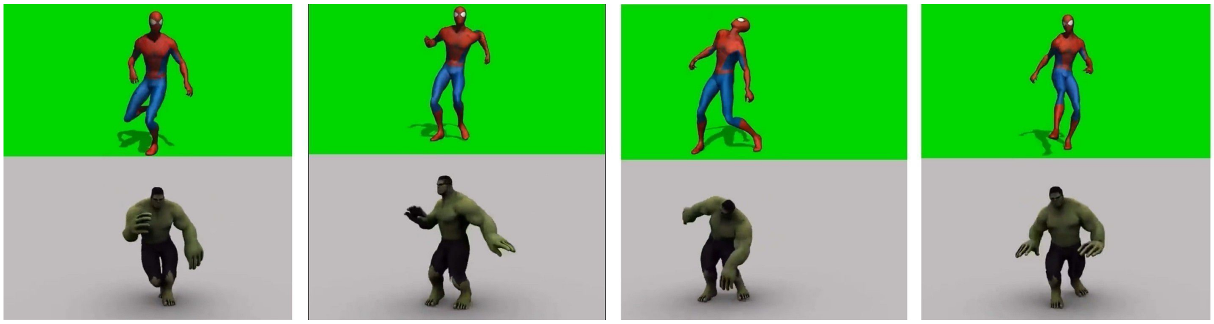


Figure 1.6. Qualitative results on Hulk (bottom) to Spiderman (top)

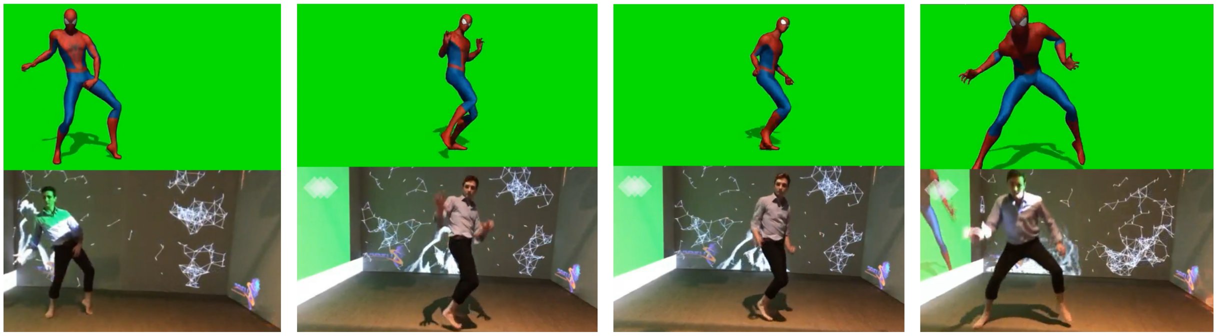


Figure 1.7. Qualitative results on Human (bottom) to Spiderman (top)

1.2 Results

For training our model we used a video of spiderman dancing [103], with a green screen background, and consists of approximately 3400 frames. Before training, the video is resized to $227 \times 227 \times 3$ and the Autoencoder is trained to extract meaningful features. We train the autoencoder for 20 epochs to make sure that the autoencoder does not learn pixel-wise reconstruction of the spiderman in the video, rather focuses on the background and generating a silhouette. For testing our system we used videos of other superheroes [103] dancing to different songs. We also recorded a few videos of one of our team mates dancing similar to spiderman which we used for testing. Querying with the basic greedy approach gave us results that are great matches in terms of the posture of the character but there was a lack of continuity when recombined into a video. Using VMO, the results were better in terms of continuity. Figures 1.6 and 1.7 shows the results that we obtained from our query mechanism.

We experimented with various thresholds and chose the one that resulted in very meaningful gestures. Sometimes, we see that the agent (spiderman) stops at a particular gesture, when it cannot find a response. This is because our system feels that for that particular query segment, jumping to any other gesture would make the cost too high. Since our model is a trade-off between a dynamic programming method and a greedy method, the model chooses to make the same gesture to try to reduce the overall cost.

Inorder to see this more clearly, we experiment with the thresholds. When the threshold is set to infinity, the model just uses VMO and when set to zero, it uses a greedy approach. If we do not have the trade-off, then the model outputs less meaningful outputs (see fig 7)

Another observation was that our model manages to achieve both continuity and meaningful responses to query, when compared to using plain VMO method. As we see in Figure 10, the top two row shows the results obtained when using VMO and the bottom two rows demonstrate the results of our model. We take three consecutive frames of the query (hulk) and observe the responses of the model (spiderman). It is clear our model makes a continuous prediction, while at the same time choosing the best possible output for the corresponding query frame.

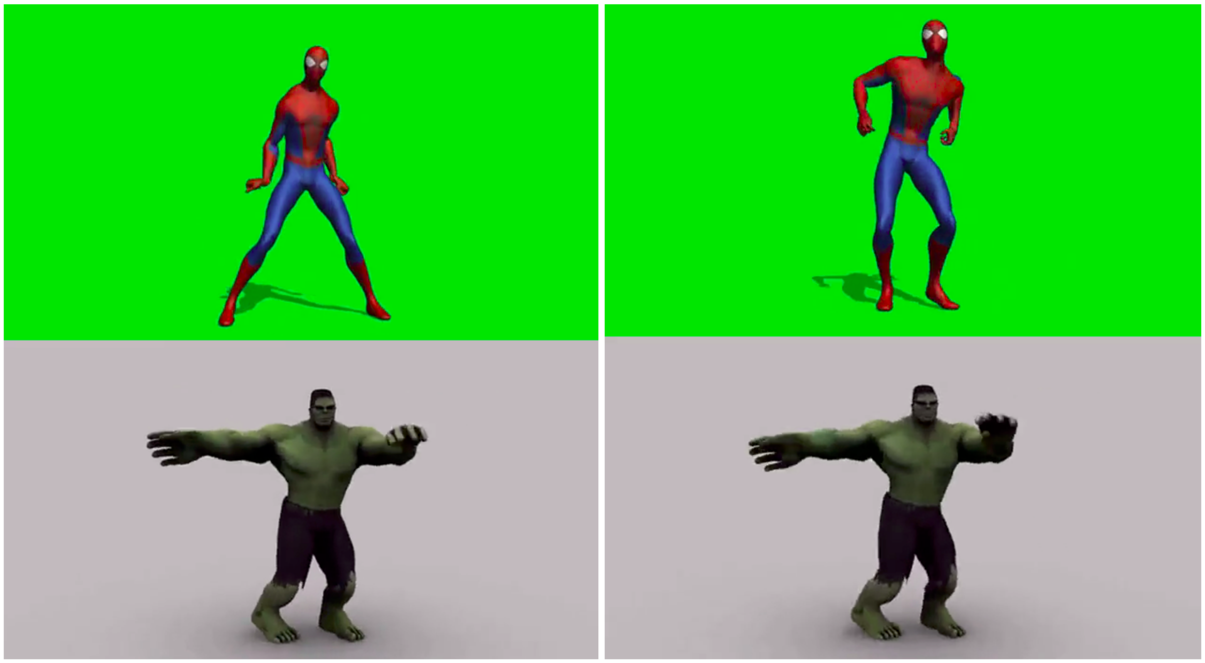


Figure 1.8. Left: Results using plain VMO, Right: Results using our model

1.3 Conclusion

In this chapter, we propose a method to perform motif detection and use those motifs to perform a ”call-and -response game” using an autoencoder to learn deep features and a Variable Markov Oracle to act as a self attention mechanism for querying. We show qualitative results of our model by using animated characters, demonstrating that recombinant query-based human movement video synthesis is possible.

1.4 Acknowledgements

Chapter 1, contains material in found in Gokul, V., Balakrishnan, G. P., Dubnov, T., & Dubnov, S. (2019, March). Semantic Interaction with Human Motion Using Query-Based Recombinant Video Synthesis. In 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 379-382). IEEE. The dissertation author was the primary investigator and author of the paper [44].

Chapter 2

Creative Improvised Interaction with Generative Musical Systems

In our previous studies we developed SOTA generative music models, with focus on interactive machine improvisation that can learn musical style from live or off-line examples and then produce ‘more of the same” [5]. This “same” was interesting in improvisation settings, since the variations maintained resemblance to the immediate expressions of the musician on stage, but were distinct enough to create interest and inspire new interaction. The problem in this setting was that it was the human who found interest in the machine imitation and changed his playing strategies by being inspired by the new machine generated materials, while the artificial agent’s generation was oblivious to the musician. In order to allow for more interactivity, special tools were introduced, such as query-based improvisation that biased the choices of the artificial music generator towards materials that had more coherence with the human musician (finding hot spots in the model memory, query-matching, a-priori scenarios and more, see [88]). Nonetheless, these modifications to the generation policy of the artificial agent that were hard-wired, are largely insufficient to capture the complexity or the subtle expressive inflections in joint multi-musician improvisations. Experiments with multiple artificial musical agents that are capable of listening or influencing each other showed that varying interaction regimes has an important effect on creating interest and prolonging the interaction into a meaningful musical form [56].

The research objective of this inter-disciplinary project is to model and enhance co-creativity as it arises in improvised musical interactions between human and artificial agents, in a spectrum of practices spanning from interacting with software agents to mixed reality involving instrumental physicality and embodiment. Such creative interaction strongly involves co-improvisation, as a mixture of more or less predictable events, reactive and planned behaviors, discovery and action phases, states of volition or idleness. Improvisation is thus at the core of this project and indeed a fundamental constituent of co-creative musicianship, as well as a fascinating anthropological lever to human interactions in general. The outline of the project unfolds as follows:

- Understanding, modelling, implementing music generative and improvised interaction as a general template for symbiotic interaction between humans and digital systems (cyber-human systems)

- Creating the scientific and technological conditions for mixed reality musical systems, based on the interrelation of creative agents and active control in physical systems.

- Achieving distributed co-creativity through complex temporal adaptation of creative agents in live cyber-human systems, articulated to field experiment in musical social sciences.

This project exploring co-creativity in many dimensions of interaction, learning and generativity is notably linked to the European REACH (Raising Co-creativity in Cyber-human Musicianship) project involving the authors.

2.1 On Co-Creativity

The psychologist Margaret Boden has given much attention to the many relations between creativity and machines [12] For her, creativity is the ability to find new, surprising and socially valuable ideas or artifacts, and can occur in three main ways: it can be combinatorial (new configurations of known materials), exploratory (discovering new paths in conceptual / mental spaces) or transformative (when the space itself is disrupted giving way to ideas that were properly inconceivable before). But what is the situation when part of the creativity is delegated to machines, when manifestations of co-creativity emerge from symbolic interactions between human and artificial agents?

In addition to the novelty / effectiveness criteria, cyber-human co-creativity is strongly felt when two features of improvisation linked to emergence [13] and non-linear dynamics [81] are identified: (1) emergence of cohesive behaviors that are not reducible to, nor explainable by the mere individual processes of agents; (2) apparition of non-linear regimes of structure formation, leading to rich musical co-evolution of forms. In our work with jazz improvisers, Bernard Lubat mentions the machine seems, in his words, to "liberate" him, perhaps from specific habits or automatisms. In other words, our inner atlases can be roamed and even modified by creative thinking, in order for the "unthought" (or the yet unthinkable) to find its way.

By producing emergent information structures as a result of cyber-human interaction, we might achieve an epistemological leap [7] beyond the difficulty of conceding creativity to artificial systems, and assess that creativity is not a state anyway, but rather a dynamical effect of interaction in a complex system, showing radical novelty as a marker of emergence [22]. By building on this epistemological boost, one would be able to model deep interactions that in turn will trigger co-creative behaviors.

2.2 Architecture of Improvising Musical Agents

The architecture of an agent in the improvisation system that we develop is shown in figure 2.1 The different elements of the system comprise of the following:

- Musical signal : stream of audio or multimedia content
- Symbolic signal : stream of quantized units (audio descriptors, musical vocabulary, latent representations
- Informed listening : the more the structures are learned, the more powerful the predictions become to help machine listening recognize musical units

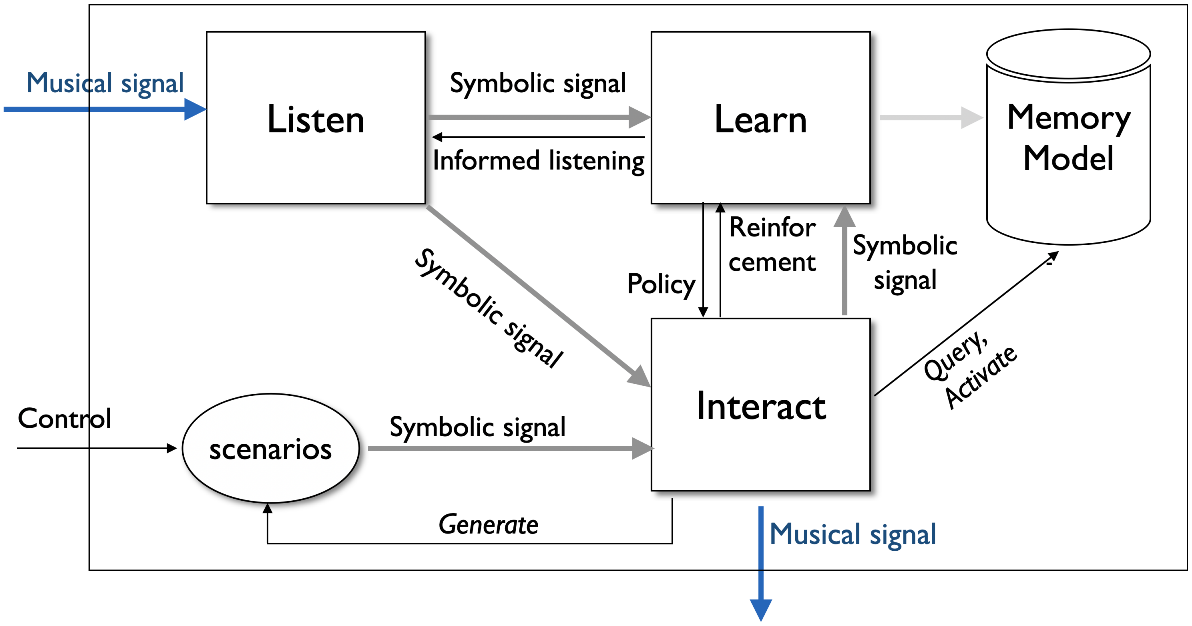


Figure 2.1. The flow of information between listen, learn and interact modules and the diverse data and control feed-backs channels in an agent

- Learn : statistical / deep modelling of musical structure and dynamics, reinforcement learning of interactive musical behaviour
- Memory model : variety of generative models for symbolic and audio signals, associated to activation states assessing the influence of the live environment on future predictions and their adequacy to musical input
- Interaction : the agent behaviour model; receive policies from the learning module ; queries the memory for generative content; assess influence from the external environment and weights on activation state of the memory; sends reinforcement signals to the learning module, follows or generate scenarios

This agent architecture can be replicated in a significant number of units in a multi-agent system, producing interaction between artificial agents as well as between agents and humans. Only musical and control signals are exchanged, with multiple cross-feedback loops (when agent A listens to agent B who listens to agent A etc.) that will promote and sustain emergence phases, such as in the Bayesian belief propagation scheme followed in [59]. Signal-symbol quantization [15], constitutes a critical part of the system as the symbolic signal constitute the main vector of information in the internal agent mechanism. Another important part is the capacity for an agent to not only ”follow” the musical input from the context, but also to respect user defined scenario, and in more extreme cases, to incrementally generate scenarios by itself as in [14], where a LSTM with bottleneck encoder - decoder and teacher forcing algorithm is used to predict the next N chords of the harmony.

2.3 Reduced Representation

Studies of human cognition suggest Rate–Distortion as a way of extracting useful or meaningful information from noisy signals [95]. The idea of reduced representation also has been recently explored in the context of representation learning in deep neural networks using a framework known as Information Bottleneck [107]. In deep learning some attempts to consider predictive information through use of a bottleneck or noisy representation in temporal models such as RNNs have recently appeared in the literature[4],[26]. Accordingly, in order to achieve a better interaction between the human and the machine, we are seeking two types of data reduction:

- lossy representation of the signal (audio or midi) that effectively reduce the dimensionality of the latent representation and allow for better generalization
- symbolization of the lossy encoding to allow for better temporal representation by using language modeling with variable memory length

Learning music representation with auto-encoder, a schematic representation of the noise induced by bit-reduction is given by Figure 2.2. Performing finite bit-size encoding and transmission of the quantized latent values from encoder Z_e to decoder Z_d is not required, since we are interested in gating and biasing the original signal towards the prior distribution by

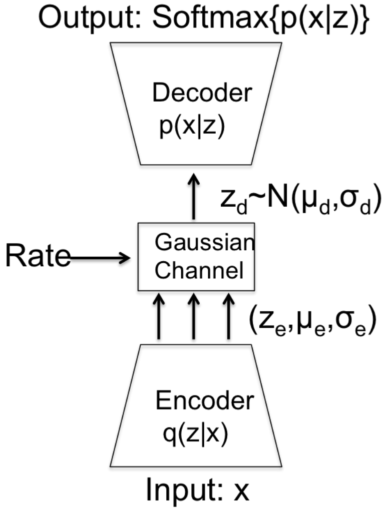


Figure 2.2. Noisy channel between encoder and decoder

encoding it at a limited bit-rate, which is given by the following optimal channel [11]

$$Q(z_d|z_e) = Normal(\mu_d, \sigma_d^2) \tag{2.1}$$

$$\mu_d = z_e + 2^{-2R}(\mu_e - z_e) \tag{2.2}$$

$$\sigma_d^2 = 2^{-4R}(2^{2R} - 1)\sigma_e^2 \tag{2.3}$$

To illustrate the effect of reduced data representation on predictive properties of music, we performed quantization at different rate for a monophonic encoding of music using a disentangled VAE training for a dataset of 14 two-part inventions composed by Johann Sebastian Bach. The MIDI files are collected from the *Complete Bach MIDI Index*[1]. Mutual information neural estimation (MINE) [10] was used to analyze the relations in time and across voices from their reduced latent representations.

From the results, we conclude that the mutual information values between conditional latent variables and predictive latent variables depend on the level of reduced representation. We find that reducing bit-allocation can effectively improve the mutual information between conditional latent variables and predictive latent variables for each scenario. For more details of

[1] http://www.bachcentral.com/midiindexcomplete.html

Table 2.1. The experiment results of controlling the rate to measure the mutual information between the conditional latent variables and the predictive latent variables. The first column shows different predictive scenarios. The left columns show the mutual information with different rates.

Scenario	R=10	R=100	R=1000	R=10000	Original
past-future	67.142	132.422	73.089	83.012	75.120
1st voice-2nd voice.	36.963	148.054	93.637	77.848	126.631
2nd voice-1st voice	61.643	91.893	104.920	91.821	82.037

this and other polyphonic and audio experiments we refer the readers to [28].

Musical Information Dynamics Assuming the music signal $X = x[n]$ is encoded into a sequence of latent representations $Z = z[n]$, with n denoting discrete time step n. We would like to algorithmically discover the sequential structure of Z, and be able to present the structures quantitatively. Music Information Dynamics (MID)[3, 30, 90] provides a theoretical framework that utilizes mutual information between past and present observations to model the predictability of the signal. The advantage of adopting MID is that it optimizes or calculates an information theoretic measurements on the input sequence Z and is agnostic of specific sequence related applications, such as motifs discovery or structure segmentation. MID was shown to be important for understanding human perception of music in terms of anticipation and predictability [3, 30].

An efficient formal method for studying MID for sequence $Z[n]$ is the Information Rate (IR) that considers the relation between the present measurement $Z = z[n]$ and it's past $\overleftarrow{Z} = z[1], z[2], \ldots, z[n], \ldots, z[N]$, formally defined as the maximum of mutual information over different quantized level of the sequence $S = Q(Z)$

$$IR(Z) = \max_{Q:S=Q(Z)} I(Q(Z), Q(\overleftarrow{Z})) \tag{2.4}$$

$$= H(S) - H(S|\overleftarrow{S}) \tag{2.5}$$

According to this measure, the maximal value of IR is obtained when the difference between the uncertainty of $H(S)$ and predictability $H(S|\overleftarrow{S})$ is at its greatest, meaning that there is a balance between variation and predictability. Quantization $Q(Z)$ is needed due to the need to detect inexact repetitions in the sequence Z, which in turn signifies the allowed level of similarities between observations in Z, or the amount of signal detail that is significant when comparing the present to the past.

2.4 Symbolization and Music Analysis using VMO

Variable Markov Oracle (VMO) [112] accepts a representation $Z = z[1], z[2], \ldots, z[N]$ and turns it into a symbolic sequence $S = s[1], s[2], \ldots, s[n], \ldots, s[N]$, with M states over a finite alphabet Σ. The labels are formed by finding suffixes in a graph structure constructed by the VMO algorithm. The VMO graph can be further used for generating new content by recombining motifs of variables length that are connected by the suffix links. Such recombination strategy assure that novel sequences have smooth transitions as they reproduce continuations that appear in the original data. One of the advantages of VMO as improvisation method is that it can be constructed quickly online during performance and does not require extensive training. Due to space consideration, we leave out the VMO construction and refer the readers to [32, 114]. The essential step in symbolization is finding a threshold with the highest MID value. The threshold θ partitions the space of features into categories that capture and represent the different sound elements by determining if the incoming $z[n]$ is similar to one of the frames following previous instances in the sequence pointed to be a suffix link from $n-1$. VMO symbolization step assigns two frames $z[i]$ and $z[j]$ the same label $s[i] = s[j] \in \Sigma$ if $||z[i] - z[j]|| \leq \theta$. In extreme cases, setting θ too low leads to VMO assigning different labels to every frame in Z and setting θ too high leads to VMO assigning the same label to every frame in Z. As a result, both extreme cases are incapable of capturing any temporal structures (repeated suffixes) of the time series. To find the optimal threshold θ, MID measure can be estimated by any predictive compression algorithm $C(\cdot)$. The compression gain over blocks of symbols is used to replace the the entropy term $H(\cdot)$

as our measure of complexity[71]

$$IR(Z) = \max_{\theta, s[n] \in \Sigma_\theta} [C(s[n]) - C(s[n]|\overleftarrow{S})]. \tag{2.6}$$

It should be noted that the alphabet out of the quantization is constructed dynamically, as new labels can be added when an input sample cannot be assigned to one of the existing clusters of samples already labeled by existing labels.

As an example of the effect of different levels of symbolization on discovery of the motif structure in different musical styles, we performed comparative analysis of several works for the flute [29] using human engineered (Chroma and MFCC) and machine learned representations (VAE). We provide a partial example of the finding in the figure 2.3. It can be seen that the Dongxiao music is characterized by much shorter motifs, which were found at much finer threshold value compared to Telemann that is characterized by longer motifs that required a coarse quantization. In a different work, a VMO based MID estimator was used to evaluate the performance of generative recurrent latent models for MIDI data. The results showed that Variational encoding, which added randomization into the latent representation of the generative model, resulted in an improved motif structure of musical generation output as it better resembled the motif statistics found in the original data compared to RNN methods that tend to overly reproduce repetitions having looping sub-sequences[38].

2.5 Acknowledgements

Chapter 2, is a reprint of material as it appears in Dubnov, S., Assayag, G., & Gokul, V. (2022, August). Creative Improvised Interaction with Generative Musical Systems. In 2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 121-126). IEEE. The dissertation author was one of the primary investigator and author of the paper [33]. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement

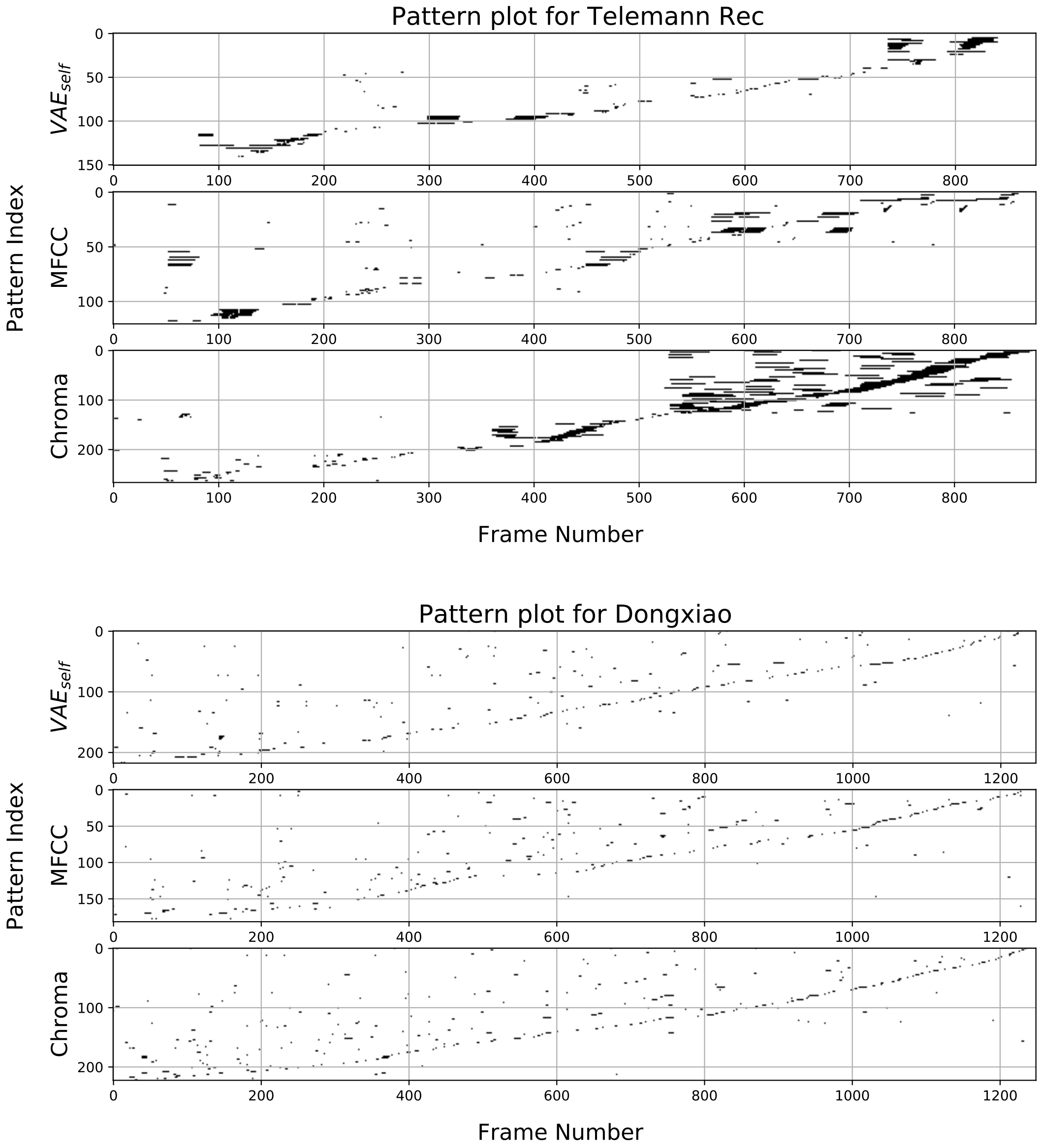


Figure 2.3. Motifs found in different flute pieces using the best VMO for VAE$_{self}$, Chroma features, and MFCCs.

#883313, project REACH : Raising Cocreativity in CyberHuman Musicianship)

Chapter 3

Switching Machine Improvisation Models by Latent Transfer Entropy Criteria

Music improvisation is the ability of musical generative systems to interact with either another music agent or a human improviser. This is a challenging task, as it is not trivial to define a quantitative measure that evaluates the creativity of the musical agent. It is also not feasible to create huge paired corpora of agents interacting with each other to train a critic system. In this chapter we consider the problem of controlling machine improvisation by switching between several pre-trained models by finding the best match to an external control signal. We introduce a measure SymTE that searches for the best transfer entropy between representations of the generated and control signals over multiple generative models.

Learning generative models of complex temporal data is a formidable problem in Machine Learning. In domains such as music, speech or video, deep latent-variable models manage today to generate realistic outputs by sampling from predictive models over a structured latent semantic space. The problem is often further complicated by the need to sample from non-stationary data where the latent features and its statistics change over time. Such situations often occur in music and audio generation, since musical structure and the type or characteristics of musical sounds change during the musical piece. Moreover, in interactive systems the outputs need to be altered so as to fit user specifications, or to match another signal that comes from the environment, which provides the context or constraint for the type of desired outcome produced by the generative system at every instance. In such cases generation by conditional sampling might be impossible due to lack of labeled training data and the need to retrain the models for each case.

We call this problem Improvisation Modeling, since it is often encountered in musical interaction with artificial musical agents that need to balance their own artificial ”creativity” with responsiveness to the overall musical context in order to create a meaningful interaction with other musicians. The ability of the artificial musical agent to make decisions and switch its responses by listening to a human improviser is important for establishing the conditions for man-machine co-creation. We consider this as a problem of controlling machine improvisation by switching between several pre-trained models by finding the best match to an external context signal. Since the match can be partially found in different generative domains, we search for best transfer entropy between reduced representations of the generated and context signals across multiple models. The added step of matching in the reduced latent space is one of the innovations of the proposed method, also motivated by theories of cognition that suggest mental representation as lossy data encoding.

In order to allow quantitative analysis of what is happening in the “musical mind”, we base our work on an information theoretic music analysis method of Music Information Dynamics (MID). MID performs structural analysis of music by considering the predictive aspects of music data, quantified by the amount of information passing from past to present in a sound recording or symbolic musical score. We extend the MID idea to include the relation between the generated and context signals and their latent representations, amounting to a total of five factors: the signal past X with its latent encoding Z, the signal present sample Y, a context signal C and its encoding into latent features T. Assuming Markov chain relations between Z-X-Y, we are looking for the smallest latent representation Z that predicts the present Y, while at the same time having maximal mutual information to the latent features T of the constraint signal. For each model we compute transfer entropy between the generated and context latent variables Z and T, respectively, and the present sample Y. It should be noted that our notion of Transfer Entropy is different from the standard definition of directed information between two random variables, since transfer entropy is estimated in the latent space of the generative model and the context signal.

We propose the use of a new metric called Symmetric Transfer Entropy (SymTE) to switch between multiple pre-trained generative models. This means that given any audio context signal, we can use SymTE to effectively switch between multiple outputs of generative models. In the chapter we will present the theory and some experimental results of switching pre-trained models according to second musical improvisation input. An important aspect of our model is eliminating the need to re-train the temporal model at each compression rate of Z since estimation of I(Y,Z) is not needed for model selection. Our assumption is that we have several pre-trained generative models(or random generators), each providing one of multiple options for improvised generation. The best model is chosen according to criteria of highest latent transfer entropy by search for the optimal reduced rate for every model, balancing between the quality of signal prediction (predicting Y from full rate Z) and matching between the past latent representation of Z and the latent representation T of the context signal for that model.

3.0.1 Causal Information

The problem of inferring causal interactions from data was formalized in terms of linear autoregression by Granger [47]. The information-theoretic notion of transfer entropy was formulated by Schreiber [101] not in terms of prediction, like in the Granger case, but in terms of reduction of uncertainty, where transfer entropy from Y to X is the degree to which Y reduced the residual uncertaintly about the future of X after the past of X was already taken into consideration. It can be shown that Granger Causality and Transfer Entropy Are Equivalent for Gaussian Variables [8] Causal entropy $\Sigma_{t=1}^{T} H(Y_t|X_{1:t}, Y_{1:t-1})$ measures the uncertainty present in the conditioned distribution of the Y variable sequence given the preceding partial X variable sequence [91].

It can be interpreted as the expected number of bits needed to encode the sequence $Y_{1:t}$ given the sequentially revealed previous Y variables and side information, $X_{1:t}$. Causal information (also known as the directed information) is a measure of the shared information between sequences of variables when the variables are revealed sequentially $\Sigma_{t=1}^{T} I(Y_t; X_{1:t} | Y_{1:t-1})$ [78]. Transfer Entropy is closely related to Causal information, except for considering the influence on Y from past of X only, not including the present instance t. Moreover, in some instances the past of X is considered for shorter past, or even just a single previous sample.

Understanding causality is important for man-machine co-creativity, especially in improvisational settings, since creating a meaningful interaction also requires answering the question of how does the human mind go beyond the data to create an experience [67]. In a way, the current work goes beyond the predictive brain hypothesis [21] to address issues of average predictability and of reduced representation of sensations as “hidden causes” or “distal causes” that maximize the communication between human and a machine in improvisational setting.

3.0.2 Estimating Transfer Entropy

Several tools and methods have been proposed to estimate transfer entropy. Refs. [97, 41] use entropy estimates based on k-nearest neighbours instead of conventional methods such as binnings to estimate mutual information. This can be extended to estimating transfer entropy, as transfer entropy can also be expressed as conditional mutual information. Similarly, methods based on Bayesian estimators [105] and Maximum Likelihood Estimation [116, 76] proposed a method to estimate transfer entropy based on Copula Entropy.

Methods using neural networks have also been proposed to estimate mutual information. Mutual Information Neural Estimator (MINE) [9] estimate mutual information by performing gradient descent on neural networks. Intrinsic Transfer Entropy Neural Estimator (ITENE) [123] proposes a two-sample neural network classifiers to estimate transfer entropy. Their method is based on variational bound on KL-divergence and pathwise estimator of Monte Carlo gradients.

Several toolboxes and plugins such as Java Information Dynamics Toolkit (JIDT) [75] provide implementations of the above mentioned methods. However, most of them have not been tested on complex high-dimensional data such as music. To our best knowledge, we are the first to propose a transfer entropy estimation method on complex data such as music and demonstrate results on tasks such as music generation.

3.1 Methodology

The main objective of our work is to calculate a metric based on transfer entropy to switch between outputs of different generative processes (say $X_1, X_2, ...X_N$), so that the output is semantically meaningful to a context signal (C). For a given X_i, we denote the past by $\bar{X_i}$ and similarly we denote the past of C as $\bar{C}$.

Transfer Entropy between two sequences is the amount of information passing from the past of one sequence to another, when the dependencies of the past of the other sequence (the sequence own dynamcis) have been already taken into account. In the case of C and model's i data X_i we have $TE_{C\to X_i} = I(X;\bar{C}|\bar{X})$ Similarly $TE_{X\to C} = I(C;\bar{X}|\bar{C})$. Writing mutual information in terms of entropy

$$I(C;\bar{X_i}) = H(C) - H(C|\bar{X_i})$$
$$I(C;\bar{X_i}|\bar{C}) = H(C|\bar{C}) - H(C|\bar{X_i},\bar{C})$$

Adding and subtracting H(C):

$$I(C;\bar{X_i}|\bar{C}) = H(C|\bar{C}) - H(C|\bar{X_i},\bar{C}) - H(C) + H(C) = I(C;\bar{X_i},\bar{C}) - I(C;\bar{C}) \tag{3.1}$$

Also:

$$I(X_i;C|\bar{X_i}) = I(X_i;\bar{C},\bar{X_i}) - I(X_i;\bar{X_i}) \tag{3.2}$$

We consider a sum of (1) and (2), let's call it symmetrical transfer entropy(SymTE):

$$SymTE = I(C;\bar{X_i}|\bar{C}) + I(X_i;\bar{C}|\bar{X_i}) \tag{3.3}$$

$$SymTE = I(C;\bar{X}|\bar{C}) + I(X_i;\bar{C}|\bar{X})$$

equals to

$$SymTE = I((C,X);\overline{(C,X)}) - I(C;X|\overline{(C,X)}) + I(C,X) - I(X_i,\bar{X}) - I(C,\bar{C})\ ,$$

where we used a notation for past of the joint pair $(\bar{C},\bar{X}_i) = \overline{(C,X_i)}$

Using the relation

I(X;—) = H(X—)-H(X—) = H(X—) - H(X) + H(X) - H(X—) = I(X;) - I(X,)

and similarily

$$I(C;\bar{X}|\bar{C}) = I(C;\bar{X}\bar{C}) - I(C,\bar{C})$$

We consider a sum of both, let's call it symmetrical TE:

$$\begin{aligned} SymTE =& I(C;\bar{X}|\bar{C}) + I(X;\bar{C}|\bar{X}) = I(C;\bar{X}\bar{C}) - I(C;\bar{C}) + I(X;\bar{X}\bar{C}) - I(X;\bar{X}) \\ &= I(C;\bar{X}\bar{C}) + I(X;\bar{X}\bar{C}) - I(C;\bar{C}) - I(X;\bar{X}) \end{aligned}$$

continuing the derivation

$$I(C;X) = H(C) + H(X) - H(C,X)$$

$$I(C;\bar{X}\bar{C}) = H(C) - H(C|\bar{X}\bar{C})$$

$$I(X;\bar{X}\bar{C}) = H(X) - H(X|\bar{X}\bar{C})$$

$$I(CX;\bar{X}\bar{C}) = H(C,X) - H(C,X|\bar{X}\bar{C}) = -I(C,X) + H(C) + H(X) - H(C,X|\bar{X}\bar{C})$$

$$= -I(C,X) + H(C) + H(X) - H(C,X|\bar{X}\bar{C}) - H(C|\bar{X}\bar{C}) + H(C|\bar{X}\bar{C}) - H(X|\bar{X}\bar{C}) + H(X|\bar{X}\bar{C})$$

$$= -I(C,X) + H(C) - H(C|\bar{X}\bar{C}) + H(X) - H(X|\bar{X}\bar{C}) - H(C,X|\bar{X}\bar{C}) + H(C|\bar{X}\bar{C}) + H(X|\bar{X}\bar{C})$$

$$= -I(C,X) + I(C,\bar{X}\bar{C}) + I(X,\bar{X}\bar{C}) + I(C,X|\bar{X}\bar{C})$$

this gives general equality:

$$I(C,\bar{X}\bar{C}) + I(X,\bar{X}\bar{C}) = I(CX,\bar{X}\bar{C}) - I(C,X|\bar{X}\bar{C}) + I(C,X)$$

plugging back to SymTE:

$$\begin{aligned} SymTE =& I(C,\bar{X}|\bar{C}) + I(X,\bar{C}|\bar{X}) \\ =& I(C,\bar{X}\bar{C}) + I(X,\bar{X}\bar{C}) - I(C,\bar{C}) - I(X,\bar{X}) \\ =& I(CX,\bar{X}\bar{C}) - I(C,X|\bar{X}\bar{C}) + I(C,X) - I(C,\bar{C}) - I(X,\bar{X}) \end{aligned}$$

One can derive the following equivalent expression

$$SymTE = I((C,X_i);\overline{(C,X_i)}) - I(C;X_i|\overline{(C,X_i)}) + I(C,X_i) - I(X_i,\bar{X}_i) - I(C,\bar{C}) \ , \quad (3.4)$$

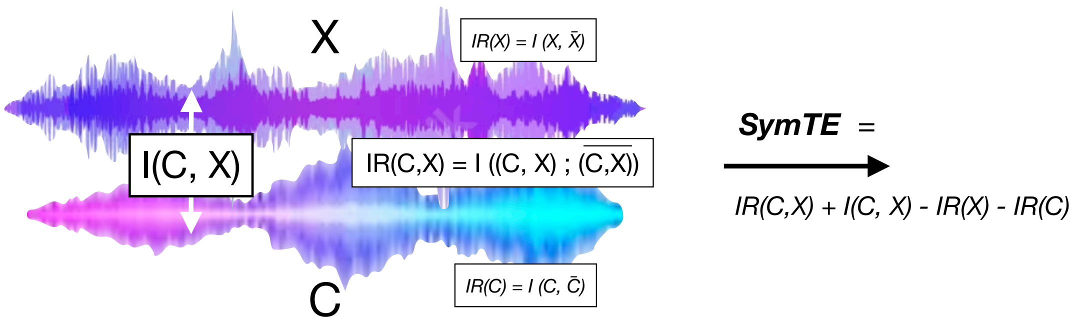
where we used a notation for past of the joint pair $(\bar{C},\bar{X}_i) = \overline{(C,X_i)}$.

The measure of mutual information between the present and the past of a signal, known as information rate (IR), will be explained in the next section. IR is commonly used in analysis of Music Information Dynamics (MID) that captures the amount of average surprisal in music signals when the next sound event is anticipated from its past. If we assume that the generation of X_i is independent of C given their joint past $\overline{(C,X_i)}$, then $I(C;X_i|\overline{(C,X_i)}) = 0$, resulting in

$$SymTE \approx I((C,X_i);\overline{(C,X_i)}) - I(X_i,\bar{X}_i) - I(C,\bar{C}) + I(C,X_i)\ , \tag{3.5}$$

which is a sum of IR of the joint pair (C,X_i) and the mutual information between C and X_i regardless of time, minus IR of the separate stream. In other words, the Symmetrical TE is a measure of surprisal present in the joint stream minus the surprisal of each of its component, plus the mutual information (lack of independence) between the individual components. In a way this captures the difference between predictive surprisal when listening to a compound stream versus surprisal when listening separately, with added component of mutual information between the voices regardless of time.

This process is schematically represented in Figure 3.1 as a combination of Information Rate and Mutual Information estimates for two musical melodies


X
IR(X) = I (X, X̄)
I(C, X)
IR(C,X) = I ((C, X) ; $\overline{(C,X)}$)
SymTE =
IR(C,X) + I(C, X) - IR(X) - IR(C)
C
IR(C) = I (C, C̄)

-0cm

Figure 3.1. Estimate of *SymTE* as a combination of Information Rate *IR* and Mutual Information *I* estimates from a generated *X* and control signal *C*

3.1.1 Predictive Surprisal Using VMO

The essential step in estimating the predictive surprisal is building a model called Variable Markov Oracle (VMO). Based on Factor Oracle (FO) string matching algorithm VMO was decveloped to allow generative improvisation for real-valued scalar or vector data, such as sequences of audio feature vectors, or data vectors extracted from human poses during dance movements. VMO uses suffix data structure for query-guided audio content generation [113] and multimedia query-matching [114, 44]. VMO operates on multivariate time serie data, *VMO* symbolizing a signal X sampled at time t, $X = x_1, x_2, \ldots, x_t, \ldots, x_T$, into a sequence $S = s_1, s_2, \ldots, s_t, \ldots, s_T$, having T states and observation frame x_t labeled by s_t. The labels are formed by following suffix links along the states in an oracle structure, whose value is one of the symbols in a finite sized alphabet Σ.

Predictive surprisal is estimated by constructing an FO automata for different threshold when search for suffix links. At each threshold value, a different oracle graph is created, and for each such oracle, a compression method of Compror (Compression Oracle) [71] algorithm C is used as an approximation to predictive information $I(X,Y) = H(Y) - H(Y|X) \approx C(Y) - C(Y|X)$. Here the entropy H is approximated by a compression algorithm C, and $C(Y) = log_2(|S|)$ is taken as the number of encoding bits for individual symbols over alphabet S, and $C(Y|X)$ is the number of bits in a block-wise encoding that recursively points to repeated sub-sequences [114].

As mentioned in the introduction, one of the advantages of using VMO for mutual information estimation is that it allows instantaneous time-varying estimates of IR based on the local information gain of encoding a signal based on linking it to its similar past. This differs from other methods of mutual information estimation like MINE that averages over the whole signal.

3.1.2 Border Cases

If $C = X$, and since $H(X,X) = H(X)$, we get $I((X,X); \overline{(X,X)}) = IR(X)$ and $SymTe = I(X,X) - IR(X) = H(X) - H(X) + H(X|\bar{X}) = H(X|\bar{X})$, which is the conditional entropy of X given its past. So TE of a pair of identical streams is its entropy rate.

If C and X are independent, $SymTE = 0$. This is based on the ideal case of IR estimator of the joint sequence $I((C,X_i); \overline{(C,X_i)})$ being able to reveal the IR of the individual sequences, and additionally capture any new emerging structure resulting from their joint occurrence.

In theory, if C and X are independent, $H(C,X) = H(C) + H(X)$, and $H(C,X|\overline{(C,X)}) = H(C|\bar{C}) + H(X|\bar{X})$, so $I((C,X_i);\overline{(C,X_i)}) = I(X_i,\bar{X_i}) + I(C,\bar{C})$. Thus, a combination of two streams may add additional information, but in practice it could be that VMO will not be able to find sufficient motifs or additional temporal structure when a mix is done. In such a case it can be that $SymTE$ estimate will become negative.

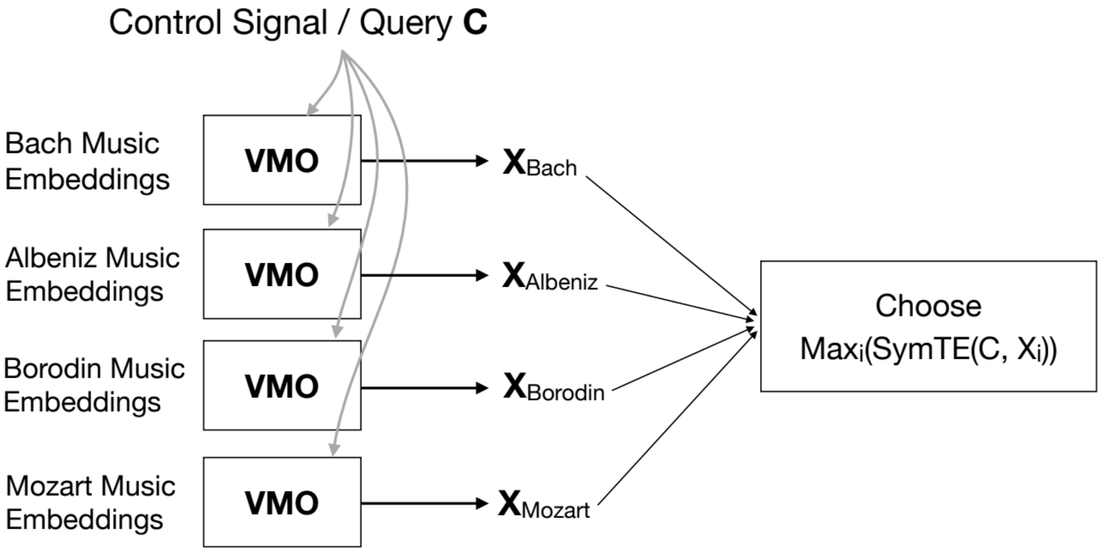
3.2 Representation Using VQ-VAE

Computing Information Rate and Mutual information for raw audio signals is an extremely challenging and computationally expensive task. We need some form of dimensionality reduction that preserves the semantic meaning of audio (style, musical rules, composer attributes etc) in the latent space. Then, we can estimate IR and MI in the lower dimensional space, quite easily. For our framework, we use a pre-trained Jukebox's Vector Quantized-Variational Autoencoder (VQ-VAE) [24, 109] to encode raw audio files to low-dimensional vectors. VQ-VAE is a type of variational autoencoder that encodes data into a discrete latent space. These discrete codes correspond to continuous vectors in a codebook. Using this, we transform our data into 8192 64-dimensional latent vectors.

3.3 Switching between Generative Models

In this section, we explain the overall workflow of our method Figure 3.2. The main objective of our method is to switch between N different generative models to match a given query C. Given training data points (musical sequences), $D_1, D_2, ..., D_N$, we compute latent representations/embeddings of each data point to get $E_1, E_2, ..., E_N$. For our method, we use the embeddings of a pretrained VQ-VAE encoder from Jukebox. We construct each generative model i as follows: 1) First we convert the query musical signal to the same latent space using Jukebox's VQ-VAE. 2) We create a VMO_i for datapoint i (in our case, we assume each datapoint represents a different composer). 3) Finally, we get the output of generative model i by querying

VMO_i with embeddings of C to get X_i, algorithm provided in [113].



-0cm

Figure 3.2. Our Methodology. Given music embeddings, we construct a VMO for each composer. For a control signal C, we query each composer's VMO to get X_i. We switch to the output X_i for a control singal C, if $SymTE(C,X_i)$ is maximum for i.

In order to choose the best output for a given query C, we calculate $SymTE(X_i,C)$ for all $i \in 1,...,N$. To calculate $SymTE(C,X_i)$, we need to calculate the individual terms of Equation 5, $I(C,\bar{C})$, $I(X,\bar{X})$, $I(C,X_i)$ and $I((C,X_i);\overline{(C,X_i)})$. To calculate $I(C,\bar{C})$, $I(X,\bar{X})$, we use VMO algorithm to create an oracle based on X_i and another oracle for C to retrieve the information rate. To calculate $I(C,X_i)$, we use MINE to calculate the mutual information between C and X_i. To calculate $I((C,X_i);\overline{(C,X_i)})$, we propose two methods, we combine both X_i and C to create a mixture, based on two methods concatenation and addition of the respective latent vectors. Then, we create an oracle for the combined (C,X_i) to calculate the IR.

3.4 Experiments and Results

We show the advantage of our method compared to other baselines by running simulations on the Labrosa APT dataset [92]. We construct a dataset with audio wav files of 4 different composers (Bach, Albeniz, Borodin, Mozart). We convert each audio file to the corresponding embeddings from a pre-trained Jukebox VQVAE [24, 109]. For X_i, we create a VMO for each composer, that can synthesize a sequence of embeddings for a given query/context signal C. For our simulations, we construct C as a segment of music (not included in the VMO construction) from any of the composers. Ideally, our SymTE measure should be high for the X_i (VMO) of the same composer C.

Table 3.1. Comparison of Accuracy and F1-Score of our methods and baselines.

Method	**Accuracy**	**F1-Score**
Random	0.22	0.17
Distance-based	0.27	0.17
Our Method (concat)	**0.44**	**0.28**
Our Method (avg)	0.36	0.21

We evaluate the effectiveness of SymTE by measuring the accuracy and F1 score. We conducted 20 trials for all the experiments. Each trial consisted of 20 query/context signals, randomly sampled from either of the 4 composers. For our baselines, we choose a random baseline and another baseline based on the euclidean distance of the embeddings, i.e choose the composer i's output, for which the euclidean distance between the embeddings of C and embeddings of X_i is the minimum. Table 3.1 shows the results of our methods and the baselines. We compare both averaging(avg) and concatenating (concat) the sequences in our experiments.We observe that our concat method achieves the best accuracy and F1-Score compared to all our baselines.

3.5 Discussion and Future Work

The methods presented in the chapter use sequence of latent vectors coming from pre-trained neural models of audio. We use VQ-VAE's embeddings and not the quantized codes, so there is only one quantization happening in this work, which is the VMO's. The reason why we chose VQ-VAE over other models is that we need strong pre-trained models and the best one currently is considered to be jukebox's VQ-VAE. Other neural models can be explored as well, as the representation is important for estimation of TE. Our query signals are 256 dimensional (≈0.71 s). Our method should work for longer queries, but the main bottleneck is the complexity of the generative model. We plan to extend this work with more elaborate results with a bigger data set and query size. We also plan to test the framework in terms of computational time, so as to enable real-time switching for music improvisation.

3.6 Acknowledgements

Chapter 3, is a reprint of material as it appears in Dubnov, S., Gokul, V., & Assayag, G. (2023, February). Switching Machine Improvisation Models by Latent Transfer Entropy Criteria. In Physical Sciences Forum (Vol. 5, No. 1, p. 49). MDPI. The dissertation author was one of the primary investigator and author of the paper [34]. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement #883313, project REACH : Raising Cocreativity in CyberHuman Musicianship)

Chapter 4

DPD-InfoGAN: Differentially Private Distributed InfoGAN

Generative Adversarial Networks (GANs) are deep learning architectures capable of generating synthetic datasets. Despite producing high-quality synthetic images, the default GAN has no control over the kinds of images it generates. The Information Maximizing GAN (InfoGAN) is a variant of the default GAN that introduces feature-control variables that are automatically learned by the framework, hence providing greater control over the different kinds of images produced. Due to the high model complexity of InfoGAN, the generative distribution tends to be concentrated around the training data points. This is a critical problem as the models may inadvertently expose the sensitive and private information present in the dataset. To address this problem, we propose a differentially private version of InfoGAN (DP-InfoGAN). We also extend our framework to a distributed setting (DPD-InfoGAN) to allow clients to learn different attributes present in other clients' datasets in a privacy-preserving manner. In our experiments, we show that both DP-InfoGAN and DPD-InfoGAN can synthesize high-quality images with flexible control over image attributes while preserving privacy.

4.1 Introduction

Deep Neural Networks can be used to train high-quality models with state-of-the-art performance in a myriad of applications, including medical image analysis, health informatics, language representation, and many more. However, building such models is not an easy task as it requires access to a large amount of high-quality data. Sharing private data is not an option in many scenarios due to regulations such as GDPR [111].

GANs [46], a class of Generative models [94, 73, 77], can be used to alleviate this arduous data-collection problem. GANs can learn the distribution of training data and generate high-quality fake data samples that have a distribution similar to the original distribution. Ideally, GANs can be used to protect the privacy of individuals in the dataset as they reveal only the distribution and not the sensitive private data of individuals. Despite this property, GANs may potentially expose the private information of training samples as they don't provide guarantees on what information the fake data may reveal about the sensitive training data. Machine learning models, including GAN models, are susceptible to a multitude of attacks including reconstruction and membership inference attacks [102, 86, 79, 49], demonstrating that additional privacy is required in the form of protecting model parameters. These attacks can be addressed through the use of differential privacy [36].

Differential privacy is the state-of-the-art model for protecting the privacy of individuals in a statistical dataset. It ensures that an adversary cannot infer if a particular individual's record is included in the dataset, hence providing the necessary guarantees to train privacy-preserving models on sensitive data. In recent times there have been studies on differentially private GANs [108, 57, 16, 42, 118]. However, most of these methods are focused on generating fixed synthetic data (with or without labels) and do not provide flexibility in controlling attributes of the synthetic data. For example, synthesizing images with different attributes (e.g. thickness, rotation, pose, etc.) involve separately training a new model with a new dataset in a private manner, which is expensive. Instead, we leverage InfoGAN [17] to facilitate control over the generated images, while preserving the privacy of the generator.

In this chapter, we propose a differentially private framework for InfoGAN and evaluate it on the MNIST dataset [69]. Our experiments show that our framework can synthesize high-quality images with strong privacy guarantees. We also analyze the trade-off between privacy and quality of control over the generated images.

Also, we propose a distributed InfoGAN (DPD-InfoGAN) with a shared Q network to capture various attributes of images owned by different clients in a privacy-preserving manner. This allows clients with limited training data to learn intricate features present in the datasets of other clients. For example, if different clients own a subset of MNIST data, then each of the clients would not be exposed to all the variances in the images (e.g. all possible rotation angles, thickness factors, etc) but will still learn to synthesize such characteristics. Aggregating such models using federated learning [62] would be an expensive process as large model parameters have to be shared and aggregated every round. To overcome this problem, our approach uses a shared Q network in a distributed setting to decrease the number of parameters exchanged and henceforth reducing communication costs.

We show that our paradigm of training distributed InfoGANs enables each client to learn rich and varied feature representations (controlling attributes of generated images) when compared with a single client setting with the same number of images.

4.2 Background

4.2.1 InfoGAN

Generative Adversarial Networks involve training two networks simultaneously: a discriminator D and a generator G. The generator maps a latent space ($p(z)$) to a fake distribution. The discriminator tries to discriminate between real data ($p(x)$) and the fake distribution. The two networks compete with each other in an adversarial setup, i.e., the generator tries to fool the discriminator into classifying its distribution as real data, while the discriminator aims to correctly classify fake and real images. This leads to a minimax game as follows:

$$\begin{aligned}\min_G \max_D V(D,G) = E_{x\sim p_{(x)}}[log(D(x))]+ \\ E_{z\sim p(z)}[log(1-D(G(x))]\end{aligned}$$

InfoGAN proposes a framework to disentangle the latent space of GANs in an unsupervised manner. The goal is to disentangle the latent space such that meaningful semantics of the data distribution are captured. The input to the generator is split into two components: noise and latent codes (from prior $p(c)$). The latent codes can be discrete or continuous. The latent codes are made meaningful by maximizing the mutual information between the generated data points and the codes. The authors of [17] use an auxiliary distribution $Q(c|x)$ to approximate the posterior, modifying the minimax game as follows:

$$\min_{G,Q} \max_{D} V_{InfoGAN}(D,G,Q) = V(D,G) - \lambda L_I(G,Q), \tag{4.1}$$

where $L_I(G,Q)$ is given by

$$L_I(G,Q) = E_{c\sim p(c),x\sim G(z,c)}[logQ(c|x)] + H(c), \tag{4.2}$$

where $H(c)$ is the entropy of the prior and is treated as constant, and λ is a hyperparameter and is set to 1. We choose p(c) as the Gaussian Distribution with zero mean and unit standard deviation. We also include continuous codes in p(c) from a uniform distribution between $[-1,1]$. p(x) refers to the real data distribution and G(z,c) refers to the output distribution of the generator G.

4.2.2 Differential Privacy

Differential privacy [36, 37] is a notion of privacy that ensures that statistical analysis does not compromise privacy by requiring that two datasets that are differing by a single individual should be statistically indistinguishable.

Definition 1. $((\varepsilon,\delta)$-Differential Privacy) A randomized mechanism $\mathscr{M}$ satisfies (ε,δ)-differential privacy ((ε,δ)-DP) when there exists $\varepsilon > 0$, $\delta > 0$, such that

$$\text{Pr}\ [\mathscr{M}(D_1) \in S] \leq e^{\varepsilon}\text{Pr}\ [\mathscr{M}(D_2) \in S] + \delta \tag{4.3}$$

holds for every $S \subseteq \text{Range}(\mathscr{M})$ and for all neighboring datasets D_1 and D_2.

Lemma 1. [118] In order to guarantee (ε, δ)-Differential privacy for the discriminator, we assign the following value to the noise scale σ_n :

$$\sigma_n = \frac{2p\sqrt{I_d log(\frac{1}{\delta})}}{\varepsilon}, \tag{4.4}$$

where the sampling probability $p = \frac{n}{N}$ (n represents the batch size, N represents the dataset size), I_d is the number of discriminator iterations for every generator iteration, ε is the privacy-loss parameter, and δ is the privacy violation parameter.

4.3 Our Approach

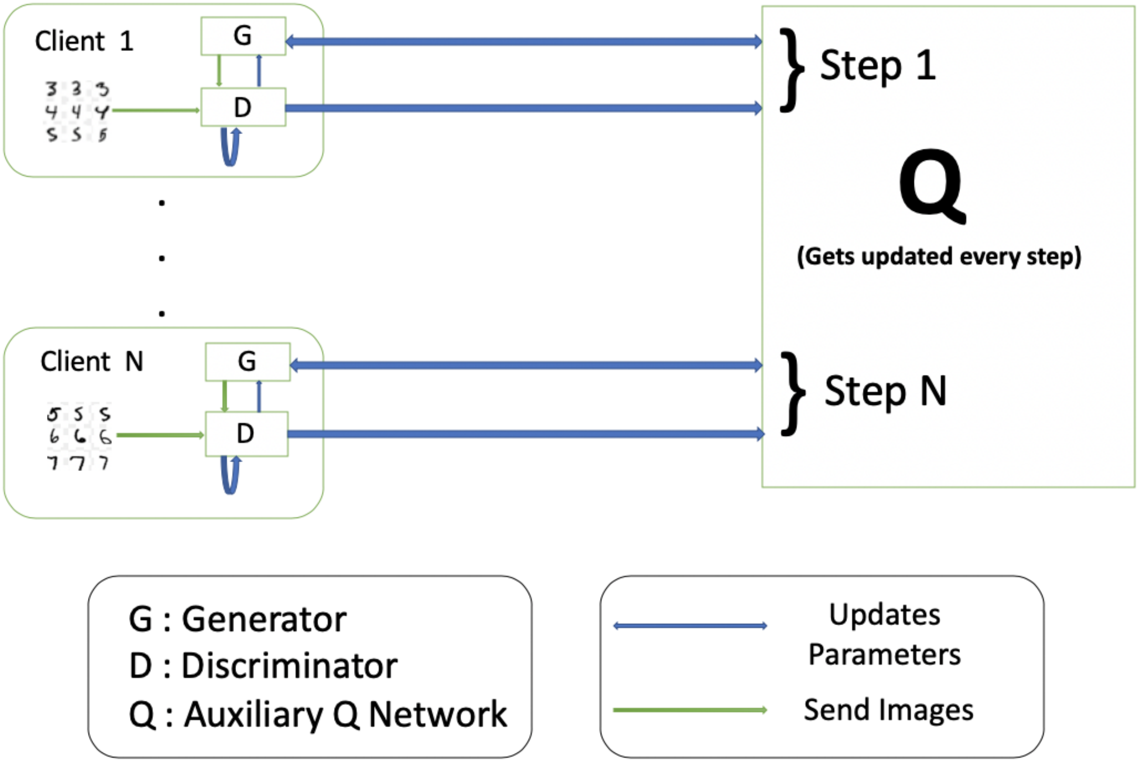


Figure 4.1. Framework for 1 Round

The details of our method to achieve a privacy-preserving InfoGAN are shown in Algorithm 4. After computing the gradients of the discriminator (line 4-7), we clip them with clipping parameter C_p (line 8) to bound the gradients. We set $I_d = 1$ in Equation 4.4 and compute noise scale σ_n. We add noise to the gradients and then update the discriminator weights using the Adam optimizer [61] (line 10). The *NLL* in line 12 refers to the Negative Log-Likelihood loss.

The training of the Q network is differentially private due to the post-processing property [37], as the Q network operates on top of the discriminator. Similarly, the generator satisfies differential privacy, as the generator receives updates from the discriminator and the Q network which are trained in a differentially private manner. We can also keep track of the privacy budget spent in our algorithm by using Moment Accountant [1] or Renyi DP accountant [80].

In the distributed setting (Figure 4.1), a similar method is used for N clients, where each client consists of a generator and a discriminator. All the clients make use of a single auxiliary network Q. As explained in Algorithm 5, the Q network is updated sequentially by each client in a given round. That is, in the same round, each client accesses the Q network that had been updated by the previous client. A single Q network is responsible for providing estimates of codes for all the clients. This reduces the communication cost as we only share the outputs of the discriminator (from client to Q-network) and the Q network (from Q-network to client), rather than sending the entire models of the generator, discriminator, and Q network as in the case of federated learning. Hence the communication load is massively reduced and therefore cost-efficient than FL.

4.4 Experiments

We ran experiments on the MNIST dataset to analyze the trade-off between privacy and the quality and semantics of generated images. The total number of training rounds R was set to 50, and each client is trained for one epoch each round. The batch size was set to 64, the number

Algorithm 4. Differentially Private InfoGAN (DP-InfoGAN)

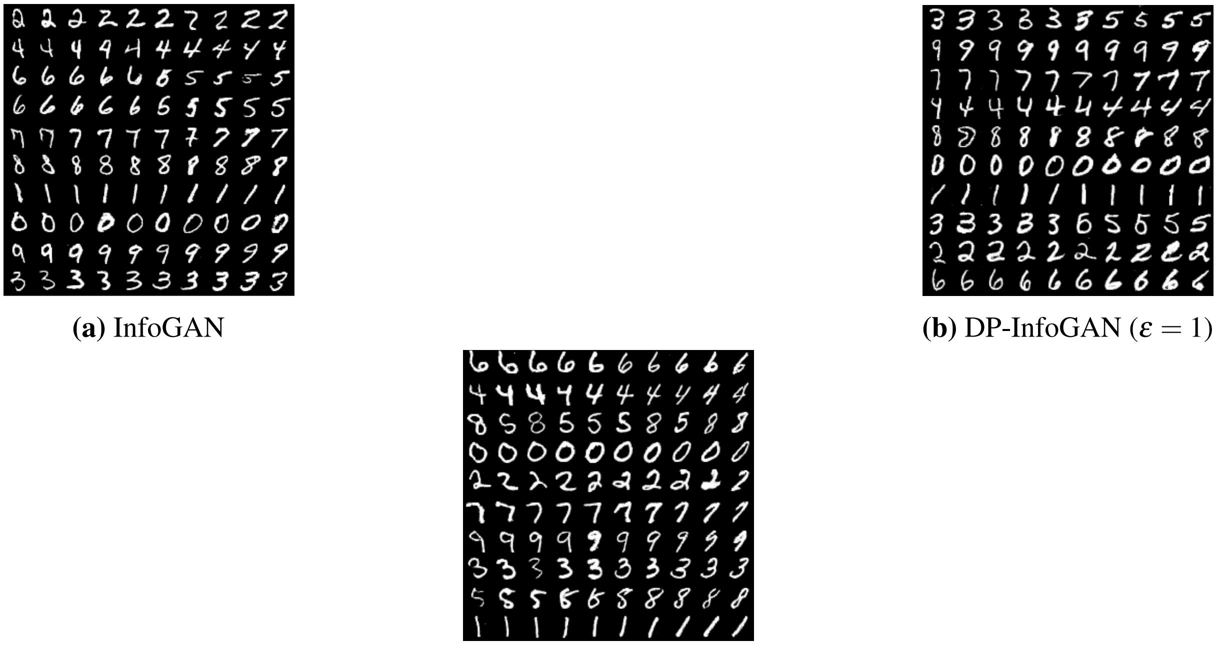
```
Clipping parameter for gradients $C_p$, Noise Scale $\sigma_n$, Discriminator $D$, Generator $G$, Auxiliary network providing estimate of the code $Q$, Real data points $X = (x_1, x_2, \dots, x_M)$, Batch size $m$, Noise prior $p(z)$, Latent code prior $p(c)$, Learning Rate $\alpha$ Differentially Private Generator $\theta_g$
Sample batch $x = \{x_i\}_{i=1}^m$ from real data points $X$
Sample noise $z = \{z_i\}_{i=1}^m$ from noise prior $p(z)$
Sample codes $c = \{c_i\}_{i=1}^m$ from prior $p(c)$
Compute batch loss for discriminator
for i = 1 to m do  $D_{loss}(x_i, z_i, c_i) := log(D(x_i)) + log(1 - D(G(z_i, c_i)))$
Calculate gradients with respect to discriminator weights $grad_d(x_i, z_i, c_i) := \nabla_{\theta_d} D_{loss}(x_i, z_i, c_i)$
Clip gradients to bound them $grad_d(x,z,c) := grad_d(x,z,c)/max(1.0, ||grad_d(x,z,c)||_2/C_p)$
Compute average gradient for batch $grad_d(x,z,c) = (1/m) * \Sigma_{i=1}^m grad_d(x_i, z_i, c_i)$

Add noise to make discriminator differentially private $grad_d(x,z,c) := grad_d(x,z,c) + (1/m) * N(0, \sigma_n^2 C^2 I)$
Update weights of the discriminator using Adam optimizer $\theta_{d_{new}} := \theta_d - \alpha.ADAM(grad_d(x,z,c), \theta_d)$
Calculate estimate of codes from Q $Q_{logits}, mean, variance = Q(D(G(z,c)))$
Compute the Negative Log Likelihood of target codes and estimate $Q_{loss} = NLL(c, mean, variance, Q_{logits})$
Compute loss and gradients for generator $G_{loss} := D(1 - log(D(G(z,c)))) + Q_{loss}$
$grad_g, grad_q := \nabla_{\theta_g} G_{loss}, \nabla_{\theta_q} G_{loss}$
$\theta_{g_{new}} := \theta_g - \alpha.ADAM(grad_g, \theta_g)$
$\theta_{q_{new}} := \theta_q - \alpha.ADAM(grad_q, \theta_q)$
return $\theta_{g_{new}}, \theta_{d_{new}}, \theta_{q_{new}}$
```

Algorithm 5. Differentially Private Distributed InfoGAN (DPD-InfoGAN)

```
Clients $C = (C_1, C_2, \dots, C_N)$, where $C_i = (G_i, D_i)$, Auxiliary network providing estimate of the code $Q$, Total number of rounds per client $R$ Differentially Private Generator $G_i$
for $r$ = 1 to $R$ do
    for $i$ = 1 to $N$ do  Train $C_i$ using Algorithm 4 $\theta_{g_{new}}, \theta_{d_{new}}, \theta_{q_{new}}$ = Algorithm1($G_i$, $D_i$, $Q$)
Update Q weights $Q$.weights = $\theta_{q_{new}}$
return $C$, $Q$
```

of epochs to 50, and δ to 10^{-5}. We fix the learning rate for the Adam optimizer to 0.0002, and sample two continuous codes from a uniform distribution between $[-1,1]$. The clipping parameter C_p is set to 1.0. We use three fractionally-strided convolutions for the generator and three convolutions for the discriminator. The Q network consists of four convolutional layers. Batch normalization is applied in all the layers. LeakyReLU is used in discriminator and Q network, while the generator uses ReLU activation.



(a) InfoGAN

(b) DP-InfoGAN ($\varepsilon = 1$)

(c) DP-InfoGAN ($\varepsilon = 0.1$)

Figure 4.2. Rotation of digits

We mainly employ qualitative evaluation as approaches such as Inception Score [99] and Frechnet Inception Distance [50] do not reflect the quality of rotation or thickness factors. First, we compare the results obtained from InfoGAN and DP-InfoGAN on a single client model. In Figure 4.3, we see that with privacy guarantees, the model has trouble differentiating between close digits (such as 3 and 5 in Figure 4.3b) but still is able to generate high-quality images. In addition, as shown in Figure 5.3b, DP-InfoGAN faces minor issues disentangling the latent space (i.e.) the results display changes in both thickness and rotation while we try to preserve only rotation of digits. Therefore, DP-InfoGAN preserves the quality of the images to an extent, but starts losing control over the attributes of images. For smaller values of ε, it becomes harder to facilitate this control. We see that in Figure 5.3c, the thickness and rotation of the digits get more entangled when compared to Figures 5.3a, 5.3b.

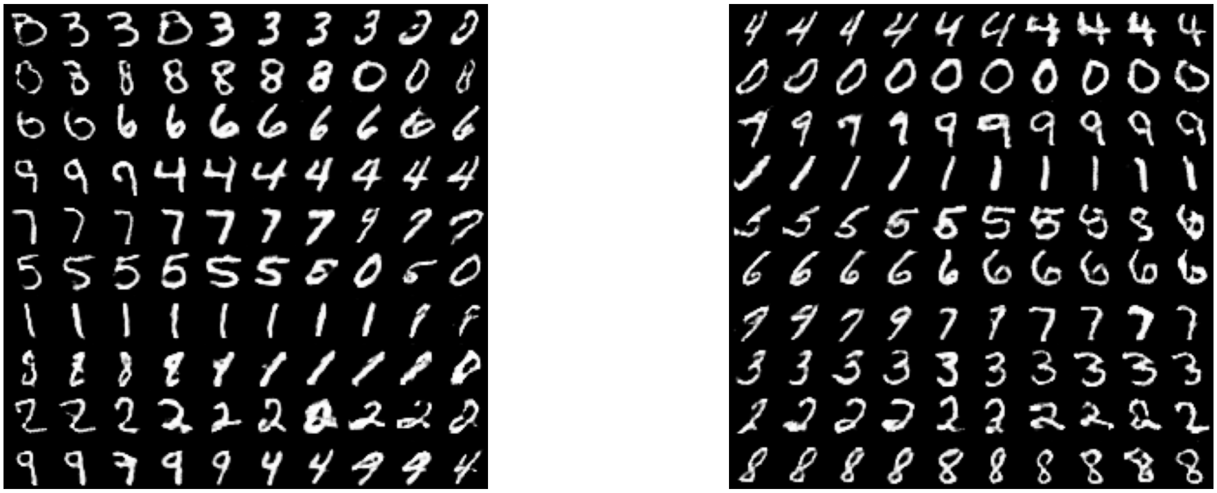
(a) InfoGAN

(b) DP-InfoGAN ($\varepsilon = 1$)

Figure 4.3. Identity of Numbers

In the distributed setting (DPD-InfoGAN), we use the same configuration and simulate experiments with each client having a subset of the MNIST data (non-overlapping). To validate our algorithm and prove that a shared Q-network can capture all possible variances in images, we run experiments with 10 clients (each having 6000 images) in a distributed setting and compare it with a single client having 6000 images. We observe that, in the distributed setting, the generated images display a varied change in thickness and rotation when compared to the non-distributed setting. In Figures 4.4b, 4.5b, we see more variety in rotation and thickness when compared to a single client setting as shown in Figures 4.4a, 4.5a. For example, digits 1 and 8 in Figure 4.4b have more variations in rotations than in Figure 4.4a. This indicates that even when a client does not have variations in its training images, it can still generate those variations as the shared Q-network is continuously updated on the clients' datasets in a privacy-preserving manner and captures all possible feature variances present in the datasets of other clients.

We ran experiments on the FashionMNIST dataset using the same setup as used for the MNIST dataset. Since the data points in the FashionMNIST dataset do not vary for rotation, we demonstrate the results for the thickness factor. In the non-distributed setting (DP-InfoGAN),

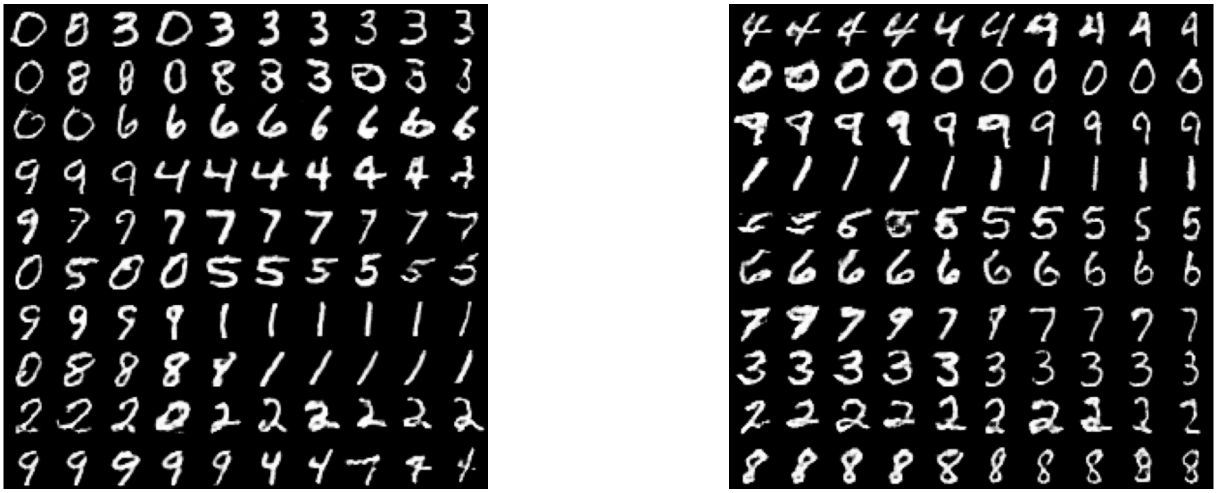


(a) Number of Clients : 1

(b) Number of Clients : 10

Figure 4.4. Rotation of digits ($\varepsilon = 10$)

we again find that as the value of epsilon decreases (more privacy), the model has trouble differentiating between objects like shoes and shirts. The amount of thickness variation also reduces as the privacy guarantees increase, as shown in Figure 4.7.



(a) Number of Clients : 1

(b) Number of Clients : 10

Figure 4.5. Thickness of digits ($\varepsilon = 10$)

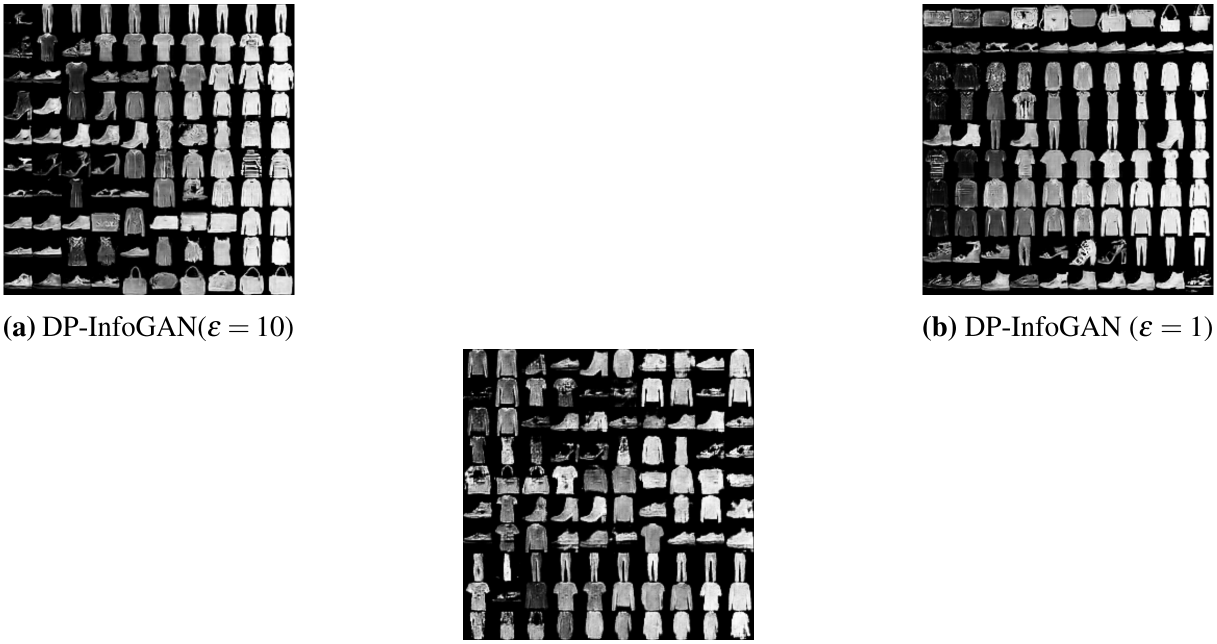
In the distributed setting (Figures 4.6a, 4.6b), we again find that with an increase in the number of clients, the shared Q-network helps in learning more variations for the thickness factor, when compared to a single client with the same number of images.

(a) Number of Clients : 1

(b) Number of Clients : 10

Figure 4.6. Variation in thickness of images for varying number of clients ($\varepsilon = 0.1$)

We mainly employ qualitative evalution as approaches such as Inception Score [99] and Frechnet Inception Distance [50] do not reflect the quality of rotation or thickness factors. These measures reflect the quality of the generated images and are not the best methods to use in our approach, as we have to evaluate the amount of rotation and thickness in addition to the quality of the images. Hence, we employ qualitative evaluation.



(a) DP-InfoGAN($\varepsilon = 10$)

(b) DP-InfoGAN ($\varepsilon = 1$)

(c) DP-InfoGAN ($\varepsilon = 0.1$)

Figure 4.7. Variation in thickness of images for varying ε

4.5 Conclusion and Future Work

In this chapter, we propose a privacy-preserving version of InfoGAN (DP-InfoGAN) that guarantees the privacy of the training data samples. Our results show that DP-InfoGAN can synthesize high-quality images with control on image attributes. Our framework can keep track of the privacy budget spent by using Moment Accountant or Renyi DP accountant. We also extend the framework to a distributed setting (DPD-InfoGAN) by using a shared Q network. We show that our training paradigm in the distributed setting captures varied image characteristics even when each client has limited data. As part of future work, we plan to explore privacy-preserving and distributed/federated versions of CycleGANs, BigGANs, etc.

4.6 Acknowledgements

Chapter 4, is a reprint of material as it appears in Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021, April). Dpd-infogan: Differentially private distributed infogan. In Proceedings of the 1st Workshop on Machine Learning and Systems (pp. 1-6). The dissertation author was one of the primary investigator and author of the paper [82].

Chapter 5

Bias-Free FedGAN: A Federated Approach to Generate Bias-Free Datasets

Federated Generative Adversarial Network (FedGAN) is an approach to train a GAN across distributed clients without clients having to share their sensitive training data. In this chapter, we propose a federated approach to generate audio samples (FedSpecGAN). We experimentally show that FedGAN and FedSpecGAN generates biased data samples under non-independent-and-identically-distributed (non-iid) settings. Also, we propose Bias-Free FedGAN and Bias-Free FedSpecGAN to generate synthetic datasets without bias. Our approach generates metadata at the aggregator using the models received from clients and retrains the federated model to achieve bias-free results for audio and image synthesis. Experimental results on audio (SC09 and MiniSpeechCommands) and image datasets (MNIST, CIFAR-10, and FairFace) validate our claims.

5.1 Introduction

Generative adversarial networks (GANs) [46], a class of generative models [94, 73], are used to generate synthetic datasets that are similar to the datasets in which they were trained on. Application of GANs include generating video from images, synthesizing audio and music, style transfer [6], improving resolution of pictures [70], creating deepfake videos with audio [64], etc. High-quality GANs are trained on large datasets. However, in most cases, data is distributed across different sources. Sharing sensitive data is not an option due to regulations such as GDPR , CCPA, and HIPPA. Hence, to train the entire population, a distributed GAN approach is required. [48] proposed a system that has a single generator and distributed discriminators. The discriminators exchange their model parameters to avoid overfitting. A distributed training paradigm using GANs that automatically learns feature-control variables was proposed by [82]. However, these approaches work only for iid data sources. The system proposed by [121][85] works for non-iid data sources. In their algorithm, individual discriminators are trained and they update a centralized generator. However, their work faced numerous communication challenges. To solve this problem, FedGAN (Federated GAN) was proposed by [93]. In FedGAN, multiple clients collaboratively train a model (generator and discriminator pair) without sharing their raw data; they share their local model weights with a trusted aggregator during each round of training; the aggregator updates the global model (generator and discriminator pair) using these weights. This process repeats for a pre-defined number of rounds or until convergence is achieved. However, the approach proposed by [93] works only for images. In this chapter, we propose a novel approach (FedSpecGAN) to generate audio samples in a federated manner. Our approach is similar to FedGAN, however, in the last federated round, the mean and standard deviation of each client's data are sent to the central aggregator for denormalization.

In addition, despite proving convergence and producing high-quality results, FedGAN generates biased data in multiple scenarios. FedGAN does not address this major issue. Though there is a myriad of solutions for addressing fairness and bias in GANs under the local setting [100, 119, 106, 55], no solutions have been proposed for solving this problem in the federated setting.

One of the main features of Federated Learning is the heterogeneity of the clients' data. This also causes biases in the results, as it is not possible to manually evaluate the bias or access the images of the clients. Some approaches [120, 54, 2] have been proposed to mitigate biases in federated learning. [2] use local and global reweighing and propose a fairness-aware regularization term in the training objective. [54] propose a double momentum gradient method and a weighting strategy based on the frequency of participation in the training process. However, these approaches do not provide a solution for federated/distributed GANs.

We consider an example scenario to emphasize the importance of this problem. Let us consider the task of speech synthesis in a federated setting. The goal of the generative model is to synthesize realistic audio that mimics the training dataset. In a federated setting, each client is unaware of the other clients' data. For example, let the first client have a training dataset that has audio samples of a female speaker. Let the other clients have datapoints of male speakers. We demonstrate that federated models trained under such settings can propagate bias and ignore the minority class data points while synthesizing audio samples. Similarly, this scenario can also be extended to facial image synthesis, where a particular client could have images of black people while other clients have images of people belonging to a different race. This is a serious issue as these models can perform poorly and unethically for minority classes.

In this chapter, we propose Bias-Free FedGAN and Bias-Free FedSpecGAN to eliminate bias in federated models. In our methods, the aggregator generates a new dataset (metadata) from each of the incoming client models and retrains the federated model. We show experimental results that the combination of averaging weights of client models and retraining on metadata helps achieve bias-free results in image synthesis.

In summary, our contributions are as follows:

- We demonstrate that FedGAN produces biased results under non-iid scenarios.
- We introduce FedSpecGAN, a federated framework for audio synthesis and show that it produces biased results for audio synthesis under non-iid scenarios
- We propose Bias-Free FedGAN and Bias-Free FedSpecGAN, frameworks to train FedGAN and FedSpecGAN in an unbiased manner to produce bias-free outputs.
- We validate our claims by running experiments on the MNIST [69], FashionMNIST [117], CIFAR-10 [65], FairFace [58], SC09, and MiniSpeechCommands [115] datasets.

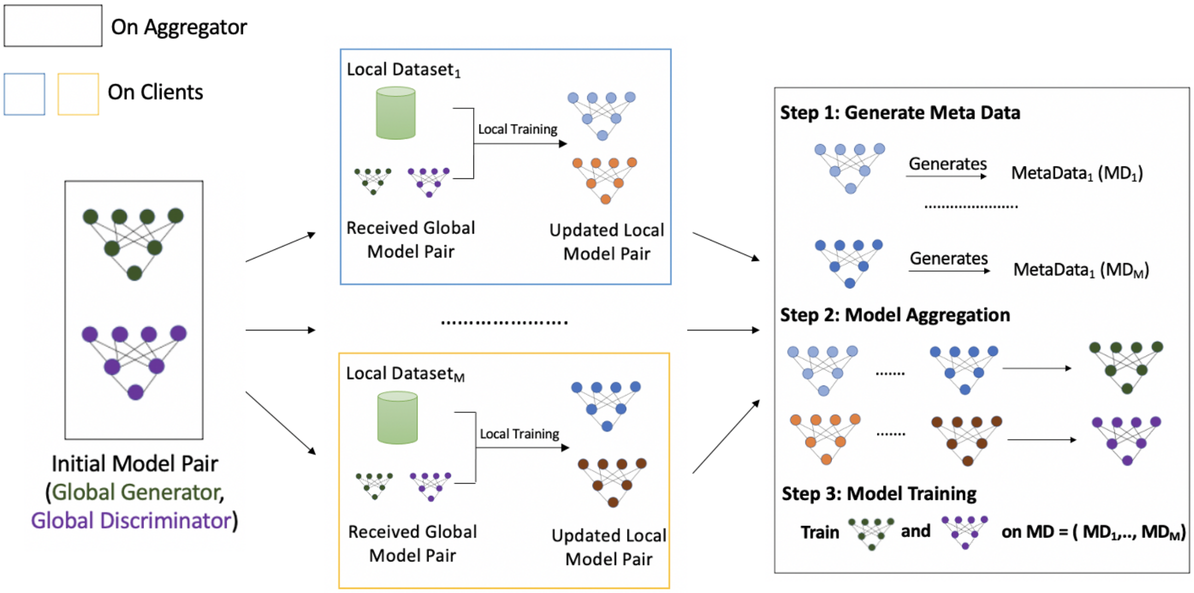


Figure 5.1. A Representative Round in Bias-Free FedGAN

5.2 Background

5.2.1 Generative Adversarial Networks

Generative Adversarial Networks employ two networks that train simultaneously together. A generator network G upsamples a latent space $p(z)$ to an image, while a discriminator D tries to predict if a given image is from the generator or the real data distribution $p(x)$. This leads to a minmax game as follows:

$$\min_G \max_D V(D,G) = E_{x\sim p_{(x)}}[log(D(x))] + \\ E_{z\sim p(z)}[log(1-D(G(x))]$$

As training progresses, the generator synthesizes images that are closely similar to the real dataset.

5.2.2 SpecGAN

SpecGAN [25] is a spectrogram-based generative model for audio synthesis. Raw audio is converted into a spectrogram using a short-time Fourier transform. The resultant spectrogram's magnitude is scaled logarithmically to align with human perception. Each frequency bin is normalized to have zero mean and unit variance. The final spectra are clipped to 3 standard deviations and rescaled to [-1,1]. SpecGAN uses the DCGAN algorithm to train the final model.

5.2.3 FedGAN

The FedGAN framework was proposed by [93]. FedGAN is used to train a GAN across non-independent-and-identically-distributed data sources. Their system uses an aggregator for averaging and broadcasting the model parameters of the generator and discriminator. In addition, the authors also prove that FedGAN has a similar performance to the general distributed GAN while reducing communication complexity.

5.3 Federated SpecGAN

In this section, we propose Federated SpecGAN (FedSpecGAN), an approach to train SpecGAN in a distributed non-iid setting. Every round, each client trains a SpecGAN and sends its model parameters to the central aggregator. In SpecGAN, each spectrogram is normalized to zero mean and unit variance before training and denormalized during inference. The aggregator does not have any data to compute the mean and standard deviation to denormalize during inference. Hence, each client sends the mean and standard deviation of their respective data to the aggregator. The aggregator computes the average of the means and standard deviations it receives from clients to denormalize during inference.

5.4 Bias-Free FedGAN and Bias-Free FedSpecGAN

In this section, we present the Bias-Free FedGAN and Bias-FreeSpecGAN algorithms in Algorithm 7. We consider *M* clients 1, 2, ..., M and *n* denote the index time. Each client *i* has a

Algorithm 6. FedSpecGAN

Number of training rounds N. Initialize local generator and discriminator for each client i: $\alpha_0^i = \alpha'$ and $\beta_0^i = \beta'$, $\forall i \in 1,2,...,M$. Learning rates of discriminator and generator, η_1 and η_2. Noise seed, z.

for n = 1 to N **do** Each client i computes local gradient $g^i(\alpha_n^i, \beta_n^i)$ from D_i and $h^i(\alpha_n^i, \beta_n^i)$ from D_i and synthetic data generated by the local generator.
Each client i updates its local model in parallel to other clients via

$$\alpha_n^i = \alpha_{n-1}^i + \eta_1 g^i(\alpha_{n-1}^i, \beta_{n-1}^i) \tag{5.1}$$

$$\beta_n^i = \beta_{n-1}^i + \eta_2 h^i(\alpha_{n-1}^i, \beta_{n-1}^i) \tag{5.2}$$

Each client i sends model parameters α_n^i and β_n^i to aggregator
Aggregator computes global generator α_n^g and global discriminator β_n^g

$$\alpha_n^g = \frac{1}{M}\sum_{j=1}^{M} \alpha_n^j; \beta_n^g = \frac{1}{M}\sum_{j=1}^{M} \beta_n^j \tag{5.3}$$

Aggregator sends α_n^g and β_n^g to clients and clients update

$$\alpha_n^i = \alpha_n^g; \beta_n^i = \beta_n^g, \forall i \in 1,2,...,M \tag{5.4}$$

Each client i sends local mean (μ_i) and standard deviation (σ_i) to aggregator. Aggregator computes mean(μ_n) and standard deviation (σ_n) for the federated model using:

$$\mu_n = \frac{1}{M}\sum_{j=1}^{M} \mu_j; \sigma_n = \frac{1}{M}\sum_{j=1}^{M} \sigma_j \tag{5.5}$$

Aggregator denormalizes generator's (α_n^g) outputs using μ_n and σ_n

local dataset D_i, generator α_n^i, and discriminator β_n^i. η_1 and η_2 denote the learning rate for the generator and discriminator respectively.

Each client runs the ADAM optimizer to train their local generators and discriminators on their datasets. Clients send their generator and discriminator model parameters to a central aggregator.

In Bias-Free FedGAN, the aggregator generates combined metadata (MD_n) using the generators received from the clients. For example, MD_n can be collected by sampling 10000 images from each client's generator. The aggregator averages α_n^i and β_n^i across i. Then, the aggregator trains the averaged model on **MD**. The aggregator sends the final generator α_n^g and discriminator β_n^g parameters to the clients. This process repeats for N rounds. To generate high-quality metadata, we start the metadata generation and training processes after a few initial rounds.

In Bias-Free FedSpecGAN, the aggregator generates and trains on metadata only during the last round to improve efficiency. Each client sends their respective mean and standard deviation to the aggregator. The aggregator computes metadata by denormalizing the outputs of each generator. Then, the aggregator normalizes the entire metadata for training using SpecGAN.

5.4.1 Why our approach works ?

We leverage the fact that each client's generator consists of an approximation of the real data distribution. We assume that each client has unbiased data. Since the aggregator has access to all the clients' trained generators, it also has access to an approximation of all the clients' data and can create metadata. The metadata consists of an equal number of images from all the generators. We retrain the federated model on a collection of all the generators' outputs to obtain a bias-free model. This process continues for each federated round.

For example, consider the scenario where client 0 has class A and clients 1 and 2 have class B (as shown in Figure **??**). The outputs of the FedGAN model would be biased towards class B. In Bias-Free FedGAN, there is an additional training step in the aggregator. The aggregator generates the metadata which would be a collection of images both belonging to class B and A in this case. The federated model is fine-tuned on the metadata before sending the model back to the clients. As the training progresses, the quality of the metadata improves and the Bias-Free FedGAN framework produces unbiased results without compromising the quality of images.

5.5 Experiments and Analysis

In this section, we show how FedGAN generates biased results under non-iid settings and how Bias-Free FedGAN and Bias-Free FedSpecGAN solve this problem. We demonstrate

Algorithm 7. Bias-Free FedGAN and
Bias-Free FedSpecGAN

```
Number of training rounds $N$. Initialize local generator and discriminator for each client $i$: $\alpha_0^i = \alpha'$ and $\beta_0^i = \beta'$, $\forall i \in 1,2,...,M$. Learning rates of discriminator and generator, $\eta_1$ and $\eta_2$. Noise seed, $z$.
for n = 1 to N do  Each client $i$ computes local gradient $g^i(\alpha_n^i, \beta_n^i)$ from $D_i$ and $h^i(\alpha_n^i, \beta_n^i)$ from $D_i$ and synthetic data generated by the local generator.
Each client $i$ updates its local model in parallel to other clients via
    $\alpha_n^i = \alpha_{n-1}^i + \eta_1 g^i(\alpha_{n-1}^i, \beta_{n-1}^i)$
    $\beta_n^i = \beta_{n-1}^i + \eta_2 h^i(\alpha_{n-1}^i, \beta_{n-1}^i)$
Each client $i$ sends model parameters $\alpha_n^i$ and $\beta_n^i$ to aggregator
Aggregator computes global generator $\alpha_n^g$ and global discriminator $\beta_n^g$
    $\alpha_n^g = \frac{1}{M} \sum_{j=1}^{M} \alpha_n^j ; \beta_n^g = \frac{1}{M} \sum_{j=1}^{M} \beta_n^j$
Bias-Free FedGAN Aggregator generates metadata $MD_n$ :
    $MD_n = [MD_n^0, MD_n^1, MD_n^2, \ldots, MD_n^M]$
    $MD_n^i = \alpha_n^i(z)$
Aggregator trains $\alpha_n^g$ and $\beta_n^g$ on metadata $MD_n$
Clients update:
    $\alpha_n^i = \alpha_n^g ; \beta_n^i = \beta_n^g, \forall i \in 1,2,...,M$
 Bias-Free FedSpecGAN  Each client i sends local mean ($\mu_i$) and standard deviation ($\sigma_i$) to aggregator. Aggregator computes mean($\mu_n$) and standard deviation ($\sigma_n$) for the federated model using:
    $\mu_n = \frac{1}{M} \sum_{j=1}^{M} \mu_j ; \sigma_n = \frac{1}{M} \sum_{j=1}^{M} \sigma_j$
Aggregator generates metadata $MD_n$ :
    $MD_n = [MD_n^0, MD_n^1, MD_n^2, \ldots, MD_n^M]$
    $MD_n^i = Denorm(\alpha_n^i(z), \mu_i, \sigma_i)$
 Aggregator trains SpecGAN $\alpha_n^g$ and $\beta_n^g$ on metadata $MD_n$
```

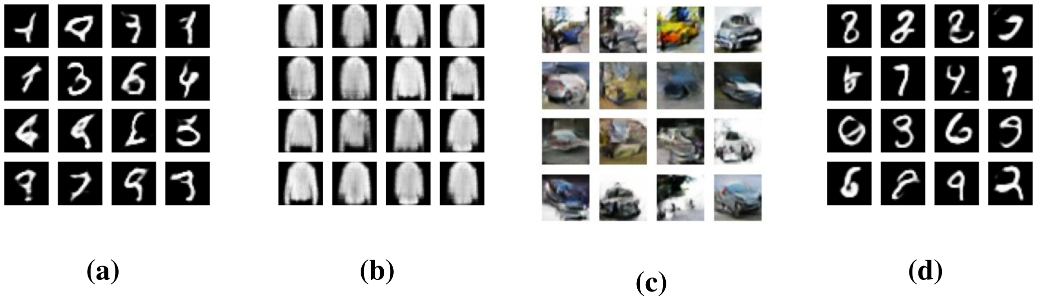


Figure 5.2. (a)FedGAN with F2 setting on MNIST; (b,c) FedGAN with F1 setting on Fashion MNIST and CIFAR-10 ; (d) FedGAN with F3 setting on MNIST

this by running experiments in a simulated setting. We use SC09, Mini Speech Commands for audio synthesis and MNIST, FashionMNIST, Cifar10, and FairFace for image synthesis. For results of FedSpecGAN, please listent to the audio samples here: https://github.com/GV1028/Bias-FreeFedGAN/

5.5.1 Inducing Bias in Client's Datasets

For all our experiments, we simulate a biased setting across 3 clients. We mainly focus on two classes for all datasets. The first client has data belonging to class 0 and all other clients have data from class 1. Every client has an equal number of data points. We denote this setting as F1.

In order to demonstrate that FedGAN in an iid setting does not generate biased results, we consider a setting, F2, where each client has an even and a balanced split of the training dataset.

In addition, we consider setting F3, where multiple minority classes exist. A single client has data points from classes 0 and 1, while other clients have images belonging to classes 2-9.

5.5.2 FedGAN and FedSpecGAN

Firstly, we show that FedGAN does not induce bias under the iid setting (F2). As shown in Figure 5.2a, we see that FedGAN generates all the classes in an iid setting for MNIST.

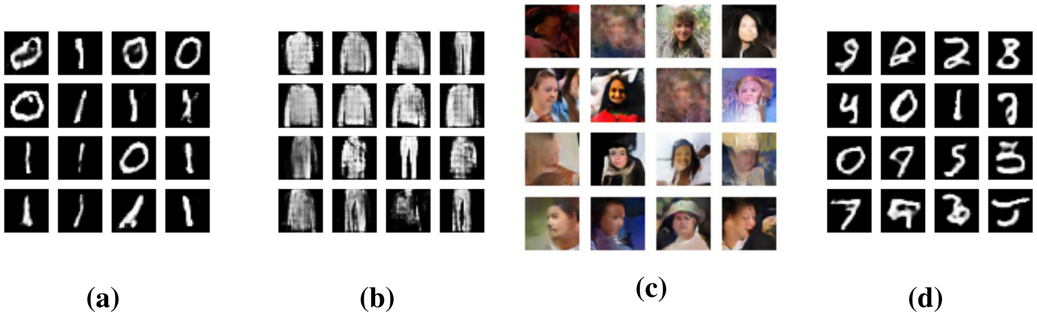


Figure 5.3. (a)FedGAN with F2 setting on MNIST; (b,c) FedGAN with F1 setting on Fashion MNIST and CIFAR-10 ; (d) FedGAN with F3 setting on MNIST

Secondly, we analyze the federated setting F1. Figures 5.2b and 5.2c show that the output of FedGAN consists of images only from the majority class ('shirt' in Fashion MNIST and 'car' in CIFAR-10). That is, the aggregated model is heavily biased towards the majority class. Similarly, for FedSpecGAN, we find that the aggregated model synthesizes audio samples of only the majority class: "eight" for SC09 and "up" for Mini Speech Commands.

In addition, in F3, we see that FedGAN (Figure 5.2d) omits classes 0 and 1 and is heavily biased towards the majority classes as it synthesizes all digits except for 0 and 1 in MNIST.

From prior experiments, we show that FedGAN and FedSpecGAN can be biased towards the majority class under non-iid settings. Referring back to the speech synthesis and face generation task, there is a high possibility that a federated model would generate male voices and white faces while ignoring minority classes.

5.5.3 Bias-Free FedGAN and FedSpecGAN

Finally, we show that Bias-Free FedGAN and Bias-Free FedSpecGAN produce unbiased results under the federated setting. For images, after each client shares the model parameters with the aggregator, the latter generates 10000 images from each client's generator to construct metadata of 50000 images. The aggregator trains an averaged model on the metadata for 100 epochs and sends the global model back to each client. In the case of audio, the aggregator generates metadata of 3000 samples only during the final round of training.

Despite the class imbalance across clients in setting F1, Bias-Free FedGAN and Bias-Free FedSpecGAN produce results that are not biased towards a particular class. Figures 5.3a,5.3b and 5.3c show the output of Bias-Free FedGAN. We see that the outputs of Bias-Free FedGAN contain data from the minority classes (number 0, class trousers, and black people). We also run experiments with multiple minority classes (classes 0 and 1) and as shown in Figure 5.3d, we can see that our solution has images from both classes 0 and 1, while FedGAN's output (Figures 5.2d) are biased towards the majority class.

5.6 Conclusion

In this chapter, we show that FedGAN generates biased outcomes under non-iid settings and provide solutions, Bias-Free FedGAN and Bias-Free FedSpecGAN, to address the same. To demonstrate the ability of our proposed solutions, we conduct experiments on the MNIST, FashionMNIST, CIFAR10, FairFace, SC09, and MiniSpeechCommands datasets. Our experiments show that Bias-Free FedGAN and Bias-Free FedSpecGAN produce fair results under non-iid settings.

5.7 Acknowledgements

Chapter 5, in part is currently being prepared for submission for publication of the material. Mugunthan, V., Gokul, V., Kagal, L., & Dubnov, S. (2021). Bias-free fedgan: A federated approach to generate bias-free datasets. arXiv preprint arXiv:2103.09876. The dissertation author was one of the primary investigator and author of the paper [83].

Chapter 6

PosCUDA: Position based Convolution for Unlearnable Audio Datasets

Deep learning models require large amounts of clean data to acheive good performance. To avoid the cost of expensive data acquisition, researchers use the abundant data available on the internet. This raises significant privacy concerns on the potential misuse of personal data for model training without authorisation. Recent works such as CUDA propose solutions to this problem by adding class-wise blurs to make datasets unlearnable, i.e a model can never use the acquired dataset for learning. However these methods often reduce the quality of the data making it useless for practical applications. We introduce PosCUDA, a position based convolution for creating unlearnable audio datasets. PosCUDA uses class-wise convolutions on small patches of audio. The location of the patches are based on a private key for each class, hence the model learns the relations between positional blurs and labels, while failing to generalize. We empirically show that PosCUDA can achieve unlearnability while maintaining the quality of the original audio datasets. Our proposed method is also robust to different audio feature representations such as MFCC, raw audio and different architectures such as transformers, convolutional networks etc.

6.1 Introduction

With the advent of large deep learning models, there is a huge need for massive clean datasets. To offset the expensive cost of data collection, researchers use the widely abundant data available on the internet. This includes image, text and audio data that individual users have uploaded to the internet. This raises an important problem on unauthorized usage of private personal data for model training. For instance, recently image generation models have been trained on facial images, artistic content etc that are available on the internet without the consent of the owners [51]. Similarly, musical audio samples of different content creators have been used for training both classification and generative models [27, 104]. All these incidents emphasize the need to create unlearnable datasets that can be uploaded by content creators while being protected from being consumed by learning models. One approach to solve this issue is by creating unlearnable datasets[40, 53, 122], i.e datasets that tamper any model's capability to learn from them. If an attacker builds a model using unlearnable datasets, the performance of such a model should be similar to a random baseline. There are three important characterstics of unlearnable datasets: 1) creation of unlearnable datasets should be fast and 3) the quality of unlearnable datasets should be as close as possible to the original data for practical purposes. Recent works such as REM [43] and CUDA [98] try to learn unlearnable datasets that satisfy all the three criteria. However REM has been shown to be vulnerable in the presence of data augmentation and CUDA affects the quality of the data by a huge factor. Moreover, all the above methods focus on image datasets and are not applicable for audio data. Audio unlearnable datasets present a unique challenge as the quality of audio can be easily detoriarated with additive noises. Furthermore, machine learning models for audio use a variety of input representations such as MFCC, spectrograms, raw audio etc that can easily break any pre-programmed unlearnability in the dataset. It is important for audio unlearnable datasets to be robust to any feature representation derived from the original dataset and data augmentations, while preserving the quality of the original audio as much as possible.

To address the above limitations, we propose PosCUDA, a method to create unlearnable datasets for audio. PosCUDA uses positional class-wise filters to help model learn relationships between labels and data, failing to generalize on new data. We apply class-wise blurs to a patch location of a sample based on a private key for each class. This means that instead of applying noise/blur over the entire data, only a small portion of the data is blurred, preserving the overall quality of the original sample. PosCUDA is fast, robust to any feature representation and minimally affects the quality of data making it applicable for practical use.

6.2 Related Work

Adversarial Poisoning: Poisoning attacks[18, 72, 74, 87] introduce some form of noise in the training process to make the model fail or mislabel classes intentionally during testing. Adversarial Poisoning has been widely used to create unlearnable datasets. However, these attacks works only for examples that have noise/trigger patterns in them. This means that during test time, the data has to have the appropriate noise and trigger patterns for adversarial poisoning to work.

Data Privacy: Data privacy methods [35, 1] preserve the model from leaking data that has been used for training[84]. Often this includes setups where multiple parties train models together sharing private data. Unlearnable datasets differ from privacy protection, as the task is to make the dataset non-usable by any model.

Unlearnable Datasets: There have been significant research in developing methods to create unlearnable datasets. [53] introduced an error-minimizing noise that reduces the training error of a class to zero, which prevents the model from learning useful features from those examples. Targeted Adversarial Poisoning (TAP) [40] use error-maximizing noises as data poisons to attack the model. Neural Tangent Generalization Attacks [122] generates label attacks to detoriorate the generarlization capability of models trained on such data. Robust Error Minmization (REM) [43] is a method to produce unlearnable datasets that are robust to adversarial training. REM uses a model to generate noise directly for the adversarial examples, rather than clean data. CUDA [98] applies a class-wise noise in the fourier domain for all the samples based on a private key for each class. This tricks the model to not learn any useful information and hence affects generalization. [39] propose a method to train a robust surrogate model and use it to generate noise for unlearnable datasets.

6.2.1 Limitations of existing works

Important characterstics of unlearnable datasets include robustness to adversarial training, high data quality and low time and compute for creation. CUDA addresses an important problem of expensive time compute needed in previous unlearnable dataset creation methods such as REM. However, due to convolving noise over the entire data sample makes CUDA not applicable for practical purposes.

Moreover, existing methods focus dominantly on image datasets. To our best knowledge, we are the first to extend the concept of unlearnability to audio datasets. Methods such as CUDA do not work well in case of temporal data, as applying noise over audio heavily corrupts the data. For example, when an user uploads an audio/music sample on the internet, they would not want to upload a noisy/jittery audio and would like to preserve the quality of the audio/music sample as much as possible. CUDA blurs out the entire audio sample leading to extremely incomprehensible audio samples.

6.3 PosCUDA

We formally motivate the problem of PosCUDA in the context of N-class classification problem.

Problem: Given a clean training dataset $DT = \{x_i, y_i\}_{i=1}^{n}$ and a clean testing dataset Dte, a performance objective $P_\theta(DT)$ denoting the performance of a classification model with parameters θ on dataset DT, our goal is to create an unlearnable dataset $\hat{DT} = \{\hat{x}_i, y_i\}_{i=1}^{n}$. The attacker trains a model with parameters $\hat{\theta}$ on $\hat{DT}$. The objective of PosCUDA is to ensure

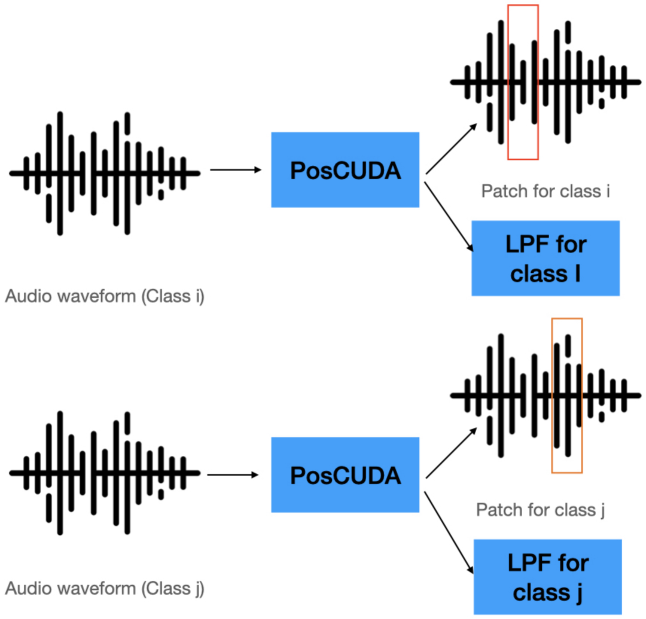


Figure 6.1. PosCUDA for Audio data: For each of the classes i,j, different patches of audio are passed through a low-pass filter unique to each class. This embeds a small class dependent positional noise in each data sample in the training set. The model learns to map these positional blurs to the labels, failing to generalize in the absence of blurs in the test dataset.

that $P_{\hat{\theta}}(Dte) \ll P_{\theta}(Dte)$ and $F(\hat{DT}, DT) \simeq 0$ is satisfied, where $F(\hat{DT}, DT)$ is a similarity score between the two datasets.

6.3.1 PosCUDA: Algorithm

In this section, we formally explain the PosCUDA algorithm to create unlearnable datasets. PosCUDA applies positional blurs to each data sample. The position of the blur and the filter is determined by a private key for each class. This means that only a small region of the data sample is affected by the blur and majority of the original quality is retained, making it usable for practical purposes. Moreover, there are multiple relationships to the label embedded in the data i.e both the noise and the position of the noise. Hence, adding a very small amount of noise should be sufficient.

For a given audio data sample belonging to class i, we first extract a small patch of the audio of patch size p. The location of the patch is specified by a parameter $pos(i)$ denoting the position to extract patch for each class i. PosCUDA uses 1-D convolutional filters f_i of size k for each class i. These filters are randomly generated for each class based on a private key. We generate these filters from $U(0,b)$ where b is the blur factor. These filters act like low-pass filters for the audio sample. The main assumption is same as that in CUDA, these keys should not be leaked. These filters are applied at the extracted patches for each data sample. We convolve 1D filters to apply these blur filters over the patch. For instance, consider an audio sample $x_i = a_1, a_2, \ldots, a_N$ belonging to class i. Let's assume the location of the patch for class i is $pos(i) = 100$, patch size $p = 80$, then we first extract the patch of size 1×80 starting at position 100, i.e $a_{100}, a_{101}, \ldots, a_{180}$ of the sample x_i. This means that only a small portion of the audio, in this case, a small patch of 1×80 gets blurred, while the remaining of the image retains the original quality.

Understanding PosCUDA: The main motivation behind PosCUDA is that majority of the machine learning models such as LSTM, Transformers or CNNs learn spatio-temporal relationships in the data. After injecting positional noise, these models are easily able to map the different blurs/noises applied at different positions to the label immediately, failing to generalize to unseen data.

Table 6.1. Test Accuracy and FAD scores of PosCUDA on SpeechCommands and FSDD datasets.

Dataset	Architecture	Clean	Ours(w/ pos)		Ours		FAD	
			b = 0.01	b = 0.3	b = 0.01	b = 0.3	b = 0.01	b = 0.3
SpeechCommands	CNN	90.52	82.30	67.33	**7.14**	7.30	5.61×10^{-4}	5.65×10^{-4}
	LSTM	90.74	84.63	12.95	18.32	**12.12**		
	Transformer	66.80	62.29	**13.50**	60.77	32.29		
FSDD	CNN	90.89	47.10	29.80	15.51	**10.04**	2.21×10^{-5}	2.25×10^{-5}
	LSTM	96.00	45.67	42.11	32.44	**22.78**		
	Transformer	91.83	72.67	53.89	62.78	**48.11**		

6.4 Experiments

In this section, we first demonstrate the effectiveness of PosCUDA on different audio datasets and various input representations. We also validate PosCUDA empirically on different classification models. We then conduct analysis on the quality of the unlearnable datasets we generate using our method. Finally, we also prove the robustness of PosCUDA under different data augmentation settings.

Datasets: We use two main datasets for our experiments:

- Speech Commands: SpeechCommands dataset consists of over 100,000 audio samples belonging to 35 classes. Each sample has approximately one second of audio.
- Free Spoken Digit Dataset: FSDD datasets consists of 6 speakers and over 3000 samples. Each samples belongs to one class between (0-9).

Architecture: It is important to test the robustness of PosCUDA across multiple architecture choices for audio classification. We also experiment with different input representations such as using raw audio or MFCC co-efficients to validate the robustness of our approach. We run experiments with the following architectures:

- M5: We use the M5 CNN architecture from [23] as one of the networks for audio classification. The M5 architecture consists of a series of 1D convolutions with batch normalization and max pooling. The model takes in raw audio waveform as input.
- LSTM: For the LSTM architecture, we extract the MFCC co-efficients for the audio. We use a one layer LSTM with 128 hidden units for our analysis.
- Transformer: We use a sequence of Encoder blocks from [110]. For our audio classification task, we use one encoder block with 8 heads and an embedding dimension of 256.

Evaluation: We evaluate the effectiveness of PosCUDA primarily on two aspects: 1) Accuracy and 2) Frechet Audio Distance (FAD)[60]. The FAD score is similar to Frechet Inception Distance [50] for images. First embeddings of both the clean audio dataset and PosCUDA's polluted dataset is computed using a pretrained model. We use PANN [63] to calculate embeddings. FAD is computed by estimating the Fréchet distance between multivariate Gaussians estimated on these embeddings. The lower the FAD score, the more close the polluted dataset is to the clean dataset, with a score of 0 meaning both the datasets are identical.

Baselines: To our best knowledge, we are the first to introduce the concept of unlearnability for audio datasets. We introduce another variant of our method without the positional blurs, i.e the low-pass filter operates on the entire data sample. This helps in higlighting the importance of positional blurs.

Implementational Details: We run every experiment for 100 epochs. We use filter size $k = 80$ and a patch size of $p = 240$ for all our experiments. The blur parameter is a hyperparemeter that can be tuned. We experiment with two settings: a low blur parameter of $b = 0.01$ and a high blur parameter of $b = 0.3$ for both the datasets. We pollute only the training set for all datasets and keep the test dataset unpolluted. We set the batch size to 128 and learning rate to 0.01.

6.4.1 Analysis

Table 6.1 shows the performance of PosCUDA compared to other baselines. PosCUDA achieves a low test score on the clean test dataset by a large margin compared to without position variant and the unpolluted dataset. For instance, PosCUDA($b = 0.01$) achieves a test accuracy of 7.12% while the clean test accuracy is 90.52 while not using positional blur achieves 82.30%. We also see that PosCUDA achieves a low test performance across a wide variety of model architectures such as CNN, LSTM and Transformers, while also being robust to different audio representations such as raw waveforms and MFCCs. An interesting observation is that Transformer architecture performs better compared to other architectures on unlearnable datasets. For example, on the FSDD dataset, the transformer architecture achieves a test accuracy of 48.11% using PosCUDA($b = 0.3$) polluted train set, while CNN and LSTM architectures achieve only 10.04% and 22.78% respectively. Transformer architectures also perform slightly better when using the without positional blur variant on the SpeechCommands dataset. Table 6.1 also shows the FAD score to compare the quality of the audio samples. As we can see, PosCUDA retains the maximum quality and achieves a very low FAD score of $2.2x10^-7$ (close to 0). This suggests that PosCUDA provides a very strong unlearnable dataset while maintaining the quality as identical to the original dataset, making it useful for practical purposes.The FAD score also slightly increases when the blur factor is increased to $b = 0.3$ but this increase is negligible (0.01×10^{-5}), again validating the importance of positional blurs. Additionally, we observe that the unlearnability efffect increases with increase in the blur parameter. For instance in both the variants, the test accuracy for FSDD decreases as the blur parameter is increased to 0.3. However, PosCUDA achieves a lower test accuracy due to the positional filters. With the same blur parameter of 0.3, PosCUDA achieves a 10.04% accuracy for CNN architecture, while non-positional variant achieves 29.80%.

Inorder to validate the hypothesis that PosCUDA maps both the position and the noise to the label, we construct a polluted test dataset for both FSDD and SpeechCommands. The polluted test dataset is constructed in the same manner as the polluted training set, i.e the same positional blurs are applied to each data sample in the test set according to its class. We observe that PosCUDA achieves a 99.8% in the test dataset under the CNN architecture. This shows that the model has succesfully learnt the mapping between the positional blurs and the corresponding classes. We also try to mix the positional blurs for each class, i.e use the positional blur corresponding to class 1 for class 2 and so on. On such a polluted dataset, the model fails again with a very low test accuracy. This further validates our claims and proves the robustness of PosCUDA. It is not easy for an attacker to randomly add noise at any position and break the unlearnability effect.

6.5 Threats

Unlearnable datasets face attacks from malicious attackers by attempting to bypass the preotection mechanisms. In this section, we discuss different threat models/scenarios for PosCUDA.

Attack 1. Attacker figures out locations of noise in the data. The attacker can then perturb data during inference at the same locations.

Defense: PosCUDA entangles both the position of the noise and the noise's private key to the label. Since we use a very small blur factor of 0.01, it is not trivial to exactly find the location of the noise. Even if the attacker figures out the location, it is not easy to replicate the original noise used. We simulated an experiment where the attacker hypothetically figures out the exact locations of the noise for each class and adds random noise to the data, but the model did not perform well on the test dataset.

Attack 2. Attacker collects the original unprotected dataset and the protected datasets to train a conditional generative model to recover the position and the noise.

Defense: It its important for the defender to delete or store the original dataset in a safe place in order to prevent access to the original data. Another strategy can be to regularly change the protection mechanism (different locations and private keys) in case the original dataset is leaked.

Attack 3: Attacker applies random blur/noise to the data to try mimic the protection of PosCUDA

Defense: Unless the attacker obtains the private key of the noise and the locations of the noise for each class label, it is not possible to mimic the protected dataset. We run experiments with random noise added to the test dataset and the model performed poorly on the test dataset.

6.6 Limitations and Future Work

While PosCUDA provides strong robust unlearnable audio datasets, the algorithm has a few limitations. PosCUDA is not ideal for long audio/musical signals, as the attacker can remove the region of noise and use the remaining data for training models. PosCUDA also works only for supervised discriminative models and the effect on unsupervised models is a good future direction to pursue. Another interesting direction would be to extend PosCUDA to generative models. Since most of the generative models such as GANs depend on some form of discriminative models, methods such as PosCUDA can be a good tool to achieve unlearnability.

6.7 Acknowledgements

Chapter 6, in part is currently being prepared for submission for publication of the material. Gokul, V., & Dubnov, S. (2024), PosCUDA: Position based Convolution for Unlearnable Audio Datasets. arXiv preprint arXiv:2401.02135. The dissertation author was the primary investigator and author of the paper [45].

Bibliography

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

[2] Annie Abay, Yi Zhou, Nathalie Baracaldo, Shashank Rajamoni, Ebube Chuba, and Heiko Ludwig. Mitigating bias in federated learning. *arXiv preprint arXiv:2012.02447*, 2020.

[3] Samer Abdallah and Mark Plumbley. Information dynamics: Patterns of expectation and surprise in the perception of music. *Connect. Sci*, 21(2-3):89–117, June 2009.

[4] Alexander Amir Alemi. Variational predictive information bottleneck. In *AABI*, 2019.

[5] Gérard Assayag, Marc Chemillier Georges Bloch, Arshia Cont, and Shlomo Dubnov. Omax brothers : a dynamic topology of agents for improvization learning. In *Workshop on Audio and Music Computing for Multimedia, ACM Multimedia*, 2006.

[6] Samaneh Azadi, Matthew Fisher, Vladimir G Kim, Zhaowen Wang, Eli Shechtman, and Trevor Darrell. Multi-content gan for few-shot font style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7564–7573, 2018.

[7] G. Bachelard. *La formation de l'esprit scientifique*. (reprint. Paris PUF coll. ≪ Quadrige ≫, 2013), Paris, Alcan, 1934.

[8] Lionel Barnett, Adam B Barrett, and Anil K Seth. Granger causality and transfer entropy are equivalent for gaussian variables. *Physical review letters*, 103(23):238701, 2009.

[9] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

[10] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, R. Devon Hjelm, and Aaron C. Courville. Mutual information neural estimation. In *Proceedings of the 35th International Conference on Machine Learning, (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 530–539. PMLR, 2018.

[11] Toby Berger. *Rate distortion theory; a mathematical basis for data compression*. Prentice-Hall Englewood Cliffs, N.J, 1971.

[12] M. Boden. Computers models of creativity. 30:23–34, 2009.

[13] C. Canonne and N. Garnier. A model for collective free improvisation. In *Mathematics and Computation in Music. Third International Conference MCM*, IRCAM, Paris, France, June 15-17 2011.

[14] Tristan Carsault, Jérôme Nika, Philippe Esling, and Gérard Assayag. Combining real-time extraction and prediction of musical chord progressions for creative applications. *Electronics, MDPI*, 10(21):2634, 2021.

[15] Axel Chemla-Romeu-Santos, Stavros Ntalampiras, Philippe Esling, Goffredo Haus, and Gérard Assayag. Cross-modal variational inference for bijective signal-symbol translation. In *Proceedings of the 22 nd International Conference on Digital Audio Effects (DAFx-19)*, 2019.

[16] Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaarfar, and Haojin Zhu. Differentially private data generative models. *arXiv preprint arXiv:1812.02274*, 2018.

[17] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[18] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

[19] Wang Cheng−i, Hsu Jennifer, and Dubnov Shlomo. Machine improvisation with variable markov oracle: Toward guided and structured improvisation. In *Computers in Entertainment (CIE) - Special Issue on Musical Metacreation, Part II*, volume 14. ACM New York, NY, USA, 2016.

[20] Wang Cheng−i and Dubnov Shlomo. The variable markov oracle: Algorithms for human gesture applications. In *IEEE Multimedia*, volume 22, pages 52–67. IEEE, 2015.

[21] Andy Clark. Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences*, 36(3):181–204, 2013.

[22] P. A. Corning. The re-emergence of ”emergence”: A venerable concept in search of a theory. *Complexity*, 7(6):18–30, 2002.

[23] Wei Dai, Chia Dai, Shuhui Qu, Juncheng Li, and Samarjit Das. Very deep convolutional neural networks for raw waveforms. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 421–425. IEEE, 2017.

[24] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[25] Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. *arXiv preprint arXiv:1802.04208*, 2018.

[26] Zhe Dong, Deniz Oktay, Ben Poole, and Alexander A. Alemi. On predictive information in rnns, 2020.

[27] Eric Drott. Copyright, compensation, and commons in the music ai industry. *Creative Industries Journal*, 14(2):190–207, 2021.

[28] S. Dubnov, K. Chen, and K. Huang. Deep music information dynamics: Novel framework for reduced neural-network music representation with applications to midi and audio analysis and improvisation. *Journal of Creative Musical Systems*, 2022.

[29] S. Dubnov, K. Huang, and C. Wang. Towards cross-cultural analysis using music information dynamics. *arXiv preprint arXiv:2111.12588*, 2021.

[30] Shlomo Dubnov. Musical information dynamics as models of auditory anticipation. In Wenwu Wang, editor, *Machine Audition: Principles, Algorithms and Systems.*, pages 371–397. IGI Global, 2011.

[31] Shlomo Dubnov, Gérard Assayag, and Arshia Cont. Audio oracle analysis of musical information rate. In *Proceedings of IEEE Semantic Computing Conference (ICSC2011)*, pages 567–571, 2011.

[32] Shlomo Dubnov, Gérard Assayag, and Arshia Cont. Audio oracle analysis of musical information rate. In *IEEE International Conference on Semantic Computing (ICSC)*, pages 567–571. IEEE, 2011.

[33] Shlomo Dubnov, Gerard Assayag, and Vignesh Gokul. Creative improvised interaction with generative musical systems. In *2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 121–126, 2022.

[34] Shlomo Dubnov, Vignesh Gokul, and Gérard Assayag. Switching machine improvisation models by latent transfer entropy criteria. In *Physical Sciences Forum*, volume 5, page 49. MDPI, 2023.

[35] Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pages 1–12. Springer, 2006.

[36] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[37] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[38] D Wright ES Koh, S Dubnov. Rethinking recurrent latent variable model for music composition. In *IEEE 20th International Workshop on Multimedia*, 2018.

[39] Bin Fang, Bo Li, Shuang Wu, Tianyi Zheng, Shouhong Ding, Ran Yi, and Lizhuang Ma. Towards generalizable data protection with transferable unlearnable examples. *arXiv preprint arXiv:2305.11191*, 2023.

[40] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. *Advances in Neural Information Processing Systems*, 34:30339–30351, 2021.

[41] Stefan Frenzel and Bernd Pompe. Partial mutual information for coupling analysis of multivariate time series. *Physical review letters*, 99(20):204101, 2007.

[42] Lorenzo Frigerio, Anderson Santana de Oliveira, Laurent Gomez, and Patrick Duverger. Differentially private generative adversarial networks for time series, continuous, and discrete open data. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 151–164. Springer, 2019.

[43] Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data against adversarial learning. *arXiv preprint arXiv:2203.14533*, 2022.

[44] Vignesh Gokul, Ganesh Prasanna Balakrishnan, Tammuz Dubnov, and Shlomo Dubnov. Semantic interaction with human motion using query-based recombinant video synthesis. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 379–382. IEEE, 2019.

[45] Vignesh Gokul and Shlomo Dubnov. Poscuda: Position based convolution for unlearnable audio datasets. *arXiv preprint arXiv:2401.02135*, 2024.

[46] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[47] Clive WJ Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, pages 424–438, 1969.

[48] Corentin Hardy, Erwan Le Merrer, and Bruno Sericola. Md-gan: Multi-discriminator generative adversarial networks for distributed datasets. In *2019 IEEE international parallel and distributed processing symposium (IPDPS)*, pages 866–877. IEEE, 2019.

[49] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. Logan: Membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies*, 2019(1):133–152, 2019.

[50] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[51] Kashmir Hill. The secretive company that might end privacy as we know it. In *Ethics of Data and Analytics*, pages 170–177. Auerbach Publications, 2022.

[52] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[53] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. *arXiv preprint arXiv:2101.04898*, 2021.

[54] Wei Huang, Tianrui Li, Dexian Wang, Shengdong Du, and Junbo Zhang. Fairness and accuracy in federated learning. *arXiv preprint arXiv:2012.10069*, 2020.

[55] Sunhee Hwang, Sungho Park, Dohyung Kim, Mirae Do, and Hyeran Byun. Fairfacegan: Fairness-aware facial image-to-image translation. *arXiv preprint arXiv:2012.00282*, 2020.

[56] Gérard Assayag. Improvising in Creative Symbolic Interaction. *Mathematical Conversations : Mathematics and Computation in Music Performance and Composition*. World Scientific; Imperial College Press, 2016.

[57] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2018.

[58] Kimmo Kärkkäinen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age. *arXiv preprint arXiv:1908.04913*, 2019.

[59] Gérard Assayag Ken Déguernel, Emmanuel Vincent. Probabilistic factor oracles for multidimensional machine improvisation. *Computer Music Journal*, 42(2):52–66, 2018.

[60] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. In *INTERSPEECH*, pages 2350–2354, 2019.

[61] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[62] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[63] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020.

[64] Pavel Korshunov and Sébastien Marcel. Deepfakes: a new threat to face recognition? assessment and detection. *arXiv preprint arXiv:1812.08685*, 2018.

[65] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[66] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[67] Thomas L Griffiths, Charles Kemp, and Joshua B Tenenbaum. Bayesian models of cognition. 2008.

[68] Hugo Larochelle and Geoffrey E Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *Advances in neural information processing systems*, pages 1243–1251, 2010.

[69] Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

[70] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[71] Arnaud Lefebvre and Thierry Lecroq. Compror: on-line lossless data compression with a factor oracle. *Information Processing Letters*, 83(1):1–6, 2002.

[72] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. *arXiv preprint arXiv:2101.05930*, 2021.

[73] Yujia Li, Kevin Swersky, and Rich Zemel. Generative moment matching networks. In *International Conference on Machine Learning*, pages 1718–1727, 2015.

[74] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 182–199. Springer, 2020.

[75] Joseph T Lizier. Jidt: An information-theoretic toolkit for studying the dynamics of complex systems. *Frontiers in Robotics and AI*, 1:11, 2014.

[76] Jian Ma. Estimating transfer entropy via copula entropy. *arXiv preprint arXiv:1910.04375*, 2019.

[77] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

[78] James Massey et al. Causality, feedback and directed information. In *Proc. Int. Symp. Inf. Theory Applic.(ISITA-90)*, pages 303–305, 1990.

[79] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 691–706. IEEE, 2019.

[80] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[81] P. Mouawad and S. Dubnov. On modeling affect in audio with non-linear symbolic dynamics. *Advances in Science, Technology and Engineering Systems Journal*, 2(3):1727–1740, 2017.

[82] Vaikkunth Mugunthan, Vignesh Gokul, Lalana Kagal, and Shlomo Dubnov. Dpd-infogan: Differentially private distributed infogan. *arXiv preprint arXiv:2010.11398*, 2020.

[83] Vaikkunth Mugunthan, Vignesh Gokul, Lalana Kagal, and Shlomo Dubnov. Bias-free fedgan: A federated approach to generate bias-free datasets. *arXiv preprint arXiv:2103.09876*, 2021.

[84] Vaikkunth Mugunthan, Vignesh Gokul, Lalana Kagal, and Shlomo Dubnov. Dpd-infogan: Differentially private distributed infogan. In *Proceedings of the 1st Workshop on Machine Learning and Systems*, pages 1–6, 2021.

[85] Vaikkunth Mugunthan, Eric Lin, Vignesh Gokul, Christian Lau, Lalana Kagal, and Steve Pieper. Fedltn: Federated learning for sparse and personalized lottery ticket networks. In *European Conference on Computer Vision*, pages 69–85. Springer, 2022.

[86] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Stand-alone and federated learning under passive and active white-box inference attacks. *arXiv preprint arXiv:1812.00910*, 2018.

[87] Tuan Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. *Advances in Neural Information Processing Systems*, 33:3454–3464, 2020.

[88] Jérôme Nika, Ken Déguernel, Axel Chemla-Romeu-Santos, Emmanuel Vincent, and Gérard Assayag. Dyci2 agents: merging the ”free”, ”reactive”, and ”scenario-based” music generation paradigms. In *International Computer Music Conference*, Shangai, China, Oct 2017.

[89] Daniil Pakhomov, Vittal Premachandran, Max Allan, Mahdi Azizian, and Nassir Navab. Deep residual learning for instrument segmentation in robotic surgery. *arXiv preprint arXiv:1703.08580*, 2017.

[90] Marcus T Pearce. Statistical learning and probabilistic prediction in music cognition: mechanisms of stylistic enculturation. *Annals of the New York Academy of Sciences*, 1423(1):378, 2018.

[91] Haim H Permuter, Young-Han Kim, and Tsachy Weissman. On directed information and gambling. In *2008 IEEE International Symposium on Information Theory*, pages 1403–1407. IEEE, 2008.

[92] Graham E Poliner and Daniel PW Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007:1–9, 2006.

[93] Mohammad Rasouli, Tao Sun, and Ram Rajagopal. Fedgan: Federated generative adversarial networks for distributed data. *arXiv preprint arXiv:2006.07228*, 2020.

[94] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[95] Chris R.Sims. Rate–distortion theory and human perception. *Cognition*, 152:181–198, 2016.

[96] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[97] Jakob Runge. Conditional independence testing based on a nearest-neighbor estimator of conditional mutual information. In *International Conference on Artificial Intelligence and Statistics*, pages 938–947. PMLR, 2018.

[98] Vinu Sankar Sadasivan, Mahdi Soltanolkotabi, and Soheil Feizi. Cuda: Convolution-based unlearnable datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3862–3871, 2023.

[99] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. arxiv 2016. *arXiv preprint arXiv:1606.03498*.

[100] Prasanna Sattigeri, Samuel C Hoffman, Vijil Chenthamarakshan, and Kush R Varshney. Fairness gan. *arXiv preprint arXiv:1805.09910*, 2018.

[101] Thomas Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000.

[102] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[103] Cool Green Studio. Playlist - superheroes dancing. Video, Jan 2017. Retrieved from https://www.youtube.com/watch?v=bjQngsn6Jq4&list=PLrzJ-NR5vXBK1LIvlMQobCA5lCxT2dctu.

[104] Bob LT Sturm, Maria Iglesias, Oded Ben-Tal, Marius Miron, and Emilia Gómez. Artificial intelligence and music: open questions of copyright law and engineering praxis. In *Arts*, volume 8, page 115. MDPI, 2019.

[105] Taiji Suzuki, Masashi Sugiyama, Jun Sese, and Takafumi Kanamori. Approximating mutual information by maximum likelihood density ratio estimation. In *New challenges for feature selection in data mining and knowledge discovery*, pages 5–20. PMLR, 2008.

[106] Shuhan Tan, Yujun Shen, and Bolei Zhou. Improving the fairness of deep generative models without retraining. *arXiv preprint arXiv:2012.04842*, 2020.

[107] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. 2015.

[108] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. Dp-cgan: Differentially private synthetic data and label generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[109] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[110] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[111] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

[112] Cheng-i Wang and Shlomo Dubnov. Guided music synthesis with variable markov oracle. In *The 3rd International Workshop on Musical Metacreation, 10th Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.

[113] Cheng-i Wang and Shlomo Dubnov. Guided music synthesis with variable markov oracle. In *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.

[114] Cheng-i Wang, Jennifer Hsu, and Shlomo Dubnov. Music pattern discovery with variable markov oracle: A unified approach to symbolic and audio representations. In *International Society for Music Information Retrieval Conference*, pages 176–182, 2015.

[115] Pete Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

[116] David H Wolpert and David R Wolf. Estimating functions of probability distributions from a finite set of samples. *Physical Review E*, 52(6):6841, 1995.

[117] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[118] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018.

[119] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Fairgan: Fairness-aware generative adversarial networks. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 570–575. IEEE, 2018.

[120] Xin Yao, Tianchi Huang, Rui-Xiao Zhang, Ruiyu Li, and Lifeng Sun. Federated learning with unbiased gradient aggregation and controllable meta updating. *arXiv preprint arXiv:1910.08234*, 2019.

[121] Ryo Yonetani, Tomohiro Takahashi, Atsushi Hashimoto, and Yoshitaka Ushiku. Decentralized learning of generative adversarial networks from multi-client non-iid data. 2019.

[122] Chia-Hung Yuan and Shan-Hung Wu. Neural tangent generalization attacks. In *International Conference on Machine Learning*, pages 12230–12240. PMLR, 2021.

[123] Jingjing Zhang, Osvaldo Simeone, Zoran Cvetkovic, Eugenio Abela, and Mark Richardson. Itene: Intrinsic transfer entropy neural estimator. *arXiv preprint arXiv:1912.07277*, 2019.

[124] Cao Zhe, Simon Tomas, Wei Shih−En, and Sheikh Yaser. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.